Mémoire présenté le :
pour l'obtention du diplôme
de Statisticien Mention Actuariat
et l'admission à l'Institut des Actuaires

Par : Florian Bouttier

Titre du mémoire :

Confidentialité : ☐ NON ☒OUI (Durée : ☐ 1 an ☒2 ans)

Les signataires s'engagent à respecter la confidentialité indiquée ci-dessus

Membres présents du jury de l'Institut des Actuaires

Entreprise :

Nom :

Signature :

Directeur de mémoire en entreprise :

Nom :

Signature :

Membres présents du jury de la filière

Invité :

Nom :

Signature :

Autorisation de publication et de mise en ligne sur un site de diffusion de documents actuariels *(après expiration de l'éventuel délai de confidentialité)*

Signature du responsable entreprise

Signature du candidat

Institut de statistiques de l'université de Paris

Construction de modèles prédictifs pour déterminer l'inflation des pièces automobiles

Auteur :
Florian Bouttier

Tuteurs entreprise :
Robin Ubezzi
Tuteur académique :
Charlotte Dion

15 décembre 2019

- Résumé -

Ce mémoire entre dans le cadre de la construction d'un outil de prédiction de l'inflation des coûts de sinistres automobiles. Nous détaillerons en particulier une méthode pour déterminer l'inflation du coût moyen des pièces automobiles. Pour cela, nous utiliserons et comparerons quatre modèles différents de séries temporelles : le modèle ARIMA, le modèle ETS, le modèle TBATS ainsi que le modèle MLP.

Afin d'obtenir des prédictions plus précises, nous construirons des variables explicatives qui nous servirons à mieux comprendre l'inflation des pièces automobiles.

Ainsi, nous commencerons par détailler la construction de la base de données et des variables explicatives. Puis nous aborderons l'aspect théorique de chacun de nos 4 modèles de séries temporelles. Finalement, nous publierons et commenterons les résultats de nos modèles.

Mots clés :
Série temporelle, *Inflation*, *ARIMA (Autoregressive integrated moving average)*, *ETS (Exponential smoothing)*, *MLP (Multi Layer Perceptron)*, *TBATS (Trigonometric Seasonal, Box-Cox Transformation, ARMA residuals,Trend and Seasonality)*, *Réseaux de neurones*, *Partitionnement en k-moyennes*

- Abstract -

This masterthesis is part of the construction of a tool for predicting automobile claims cost inflation. In particular, we will detail a method for determining the inflation of the average cost of automotive parts. To do this, we will use and compare four different time series models : the ARIMA model, the ETS model, the TBATS model and the MLP model.

In order to obtain more accurate predictions, we will construct explanatory variables that will be used to better understand auto parts inflation.

Thus, we will start by detailing the construction of the database and the explanatory variables. Then we will discuss the theoretical aspect of each of our 4 time series models. Finally, we will publish and comment on the results of our models.

Keywords :
Time series, *Inflation*, *ARIMA (Autoregressive integrated moving average)*,
ETS (Exponential smoothing), *MLP (Multi Layer Perceptron)*,
TBATS (Trigonometric Seasonal, Box-Cox Transformation, ARMA residuals,Trend and Seasonality),
Neural networks, *k-means clustering*

- Remerciements

Je souhaite tout d'abord adresser mes remerciements à Robin Ubezzi, mon tuteur en entreprise, pour son accompagnement permanent, ses apports intellectuels et ses conseils.

Je remercie également le reste de l'équipe Études indemnisation pour l'ensemble de leurs conseils et encouragements.

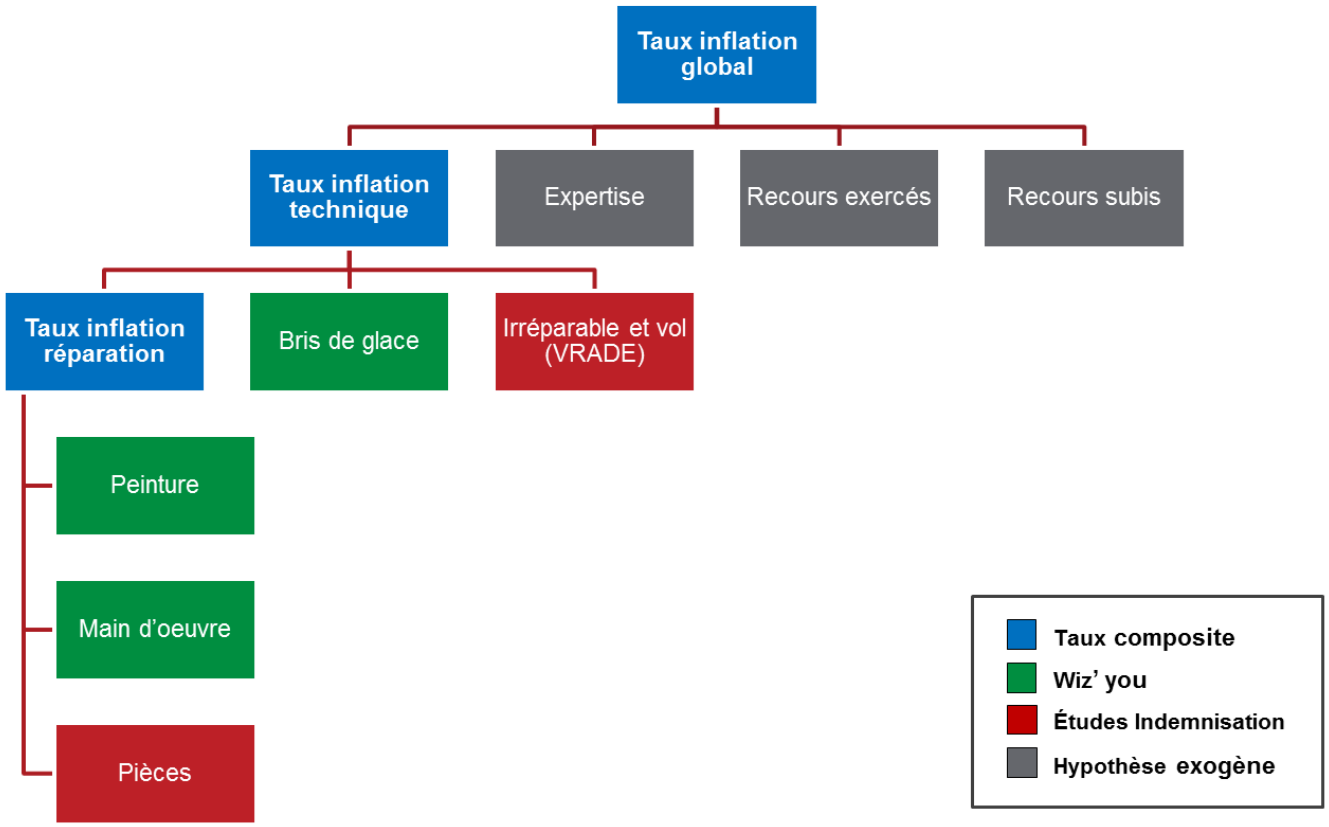
Pour la qualité des échanges que nous avons eu l'occasion d'avoir dans le cadre du projet Indicateur prospectif, je remercie également l'ensemble de l'équipe Wiz'you.

J'aimerais aussi remercier Charlotte Dion pour ses conseils et ses remarques.

Enfin, pour la relecture de ce mémoire, mais aussi plus généralement pour tout ce qu'ils ont pu m'apporter dans la vie, j'aimerais remercier mes parents.

- Note de synthèse -

Afin d'anticiper le coût des sinistres automobiles, Generali a décidé de lancer le projet "Indicateur prospectif" dans le but d'évaluer l'évolution des coûts liés à la réparation automobile. L'étude de l'inflation du coût des sinistres automobiles se divise en plusieurs postes, pilotés par deux équipes de Generali : l'équipe Études indemnisation basé à Saint-Denis et l'équipe Wiz'you, localisée à Nantes. Les postes sont répartis comme ceci :



Nous nous concentrons, dans la suite de ce mémoire, sur le poste pièces. Notre travail portera donc sur l'étude de l'inflation des pièces automobiles dans le cadre d'une réparation. L'objectif final du mémoire sera de prédire le coût moyen des pièces pour l'année 2019.
Pour ce faire, nous disposerons d'une base de données constituée de la date du sinistre, du coût total des pièces, ainsi que de diverses variables explicatives.
Notre étude utilisera et comparera 4 modèles différents de séries temporelles, dont les principes théoriques sont détaillés dans la partie *modèle* :

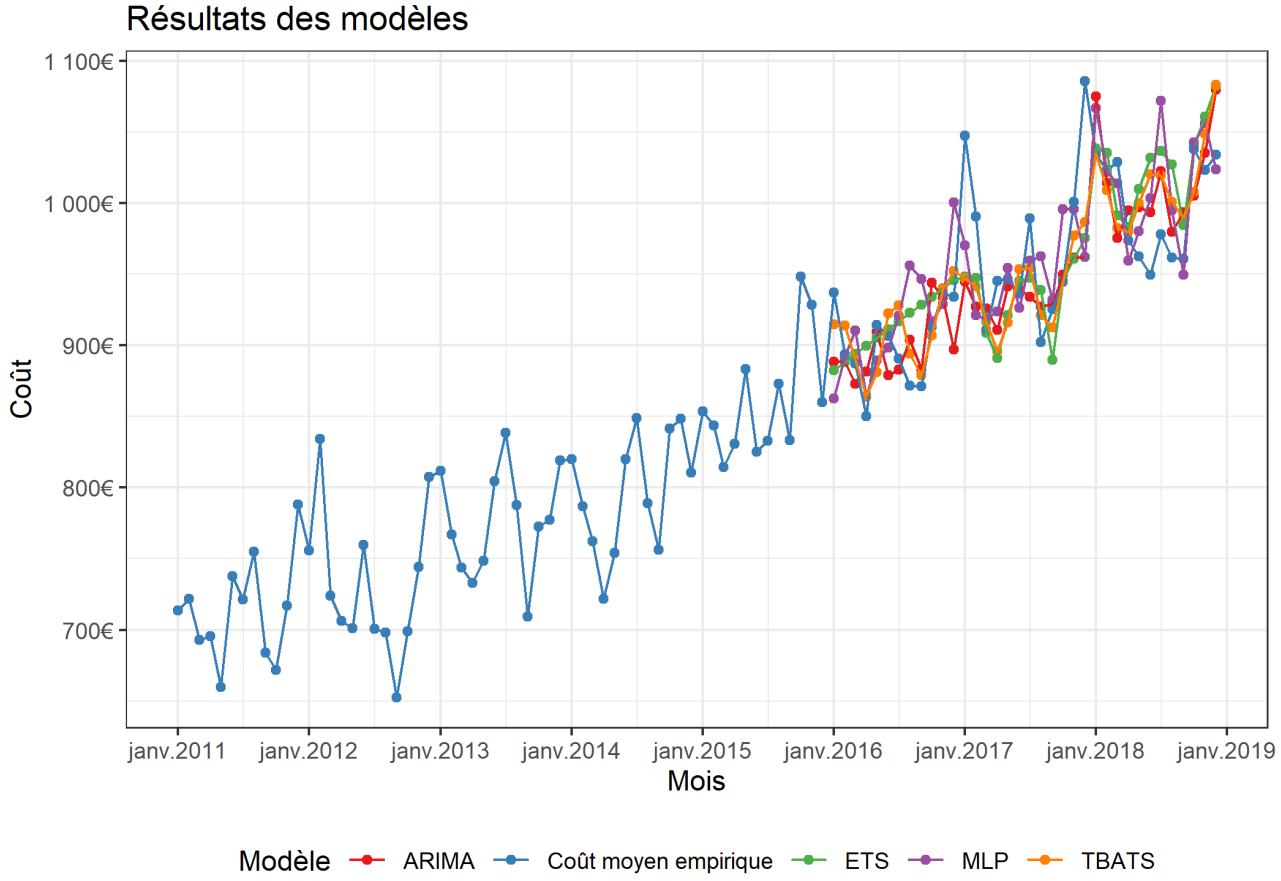
— Le modèle ARIMA (Autoregressive integrated moving average)
— Le modèle ETS (Exponential smoothing)
— Le modèle TBATS (Trigonometric Seasonal, Box-Cox Transformation, ARMA residuals,Trend and Seasonality)
— Le modèle MLP (Multi Layer Perceptron)

Ces modèles prennent en entrée une série temporelle, c'est-à-dire une suite de valeurs ordonnée, et en ressortent une autre série temporelle, prédisant la suite de la série temporelle d'entrée. Une première méthode pour prédire l'inflation est :

1. D'établir le coût moyen mensuel des pièces.
2. De construire les différents modèles sur cette série temporelle test.
3. D'établir les prédictions des modèles sur les 12 mois suivant la fin de série temporelle test
4. De construire la prédiction du coût moyen annuel en faisant la moyenne pondérée des coûts moyens mensuels prédis sur l'année en question (cette pondération s'effectue avec la répartition du nombre de sinistres par mois)

Afin de tester nos modèles, nous construisons les prédictions sur :
— L'année 2016 en prenant comme base de données le coût moyen des pièces de 2011 à 2015
— L'année 2017 en prenant comme base de données le coût moyen des pièces de 2011 à 2016
— L'année 2018 en prenant comme base de données le coût moyen des pièces de 2011 à 2017

Nous obtenons les prédictions suivantes :



Afin de comparer l'efficacité de nos modèles, nous construisons, pour chaque mois de prédiction et chaque modèle, l'erreur absolue, puis nous moyennons ces erreurs. Nous déterminons aussi l'erreur maximum pour chaque modèle.

Ces valeurs sont regroupées dans le tableau ci-dessous :

Modèle	**Écart absolu moyen**	**Écart absolu moyen en pourcentage**	**Erreur maximum**	**Erreur maximum en pourcentage**
ARIMA	31.07€	3.25%	123.23€	11.4%
ETS	34.13€	3.55%	109.93€	10.1%
TBATS	28.83€	2.99%	100.46€	9.59%
MLP	33.45€	3.47%	123.41€	11.4%

Nous remarquons que le modèle TBATS est le meilleur sur ces séries temporelles test, que ce soit concernant les écarts absolus moyens, mais aussi de l'erreur maximum.
Afin de prédire le coût moyen des pièces par année, nous allons effectuer la moyenne pondérée de chaque estimation mensuelle. Cette pondération s'effectue avec la répartition du nombre de sinistres par mois.

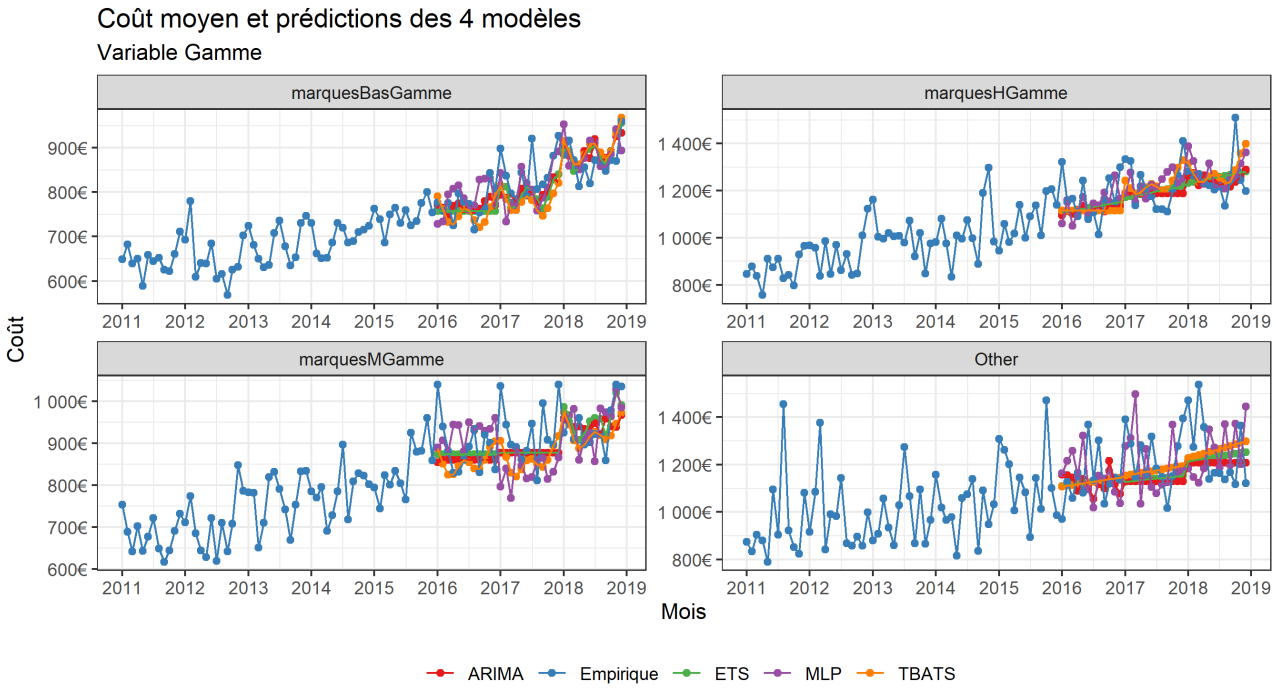
Pour la période de 2011 à 2015, La répartition du nombre de sinistres par mois est la suivante :

Janvier	Février	Mars	Avril	Mai	Juin	Juillet	Août	Septembre	Octobre	Novembre	Décembre
7.94%	7.07%	7.71%	8.05%	8.50%	9.82%	8.95%	7.63%	8.61%	8.72%	8.44%	8.55%

Ainsi en recomposant le coût moyen annuel prédit à l'aide de ces répartitions, nous obtenons les résultats suivants :

		Résultats des modèles			
Année	Coût moyen empirique	ARIMA	ETS	TBATS	MLP
2016	901.32€	897.34€	914.93€	907.76€	916.02€
2017	969.98€	938.34€	935.25€	939.53€	952.23€
2018	996.02€	1013.79€	1027.15€	1015.29€	1015.32€

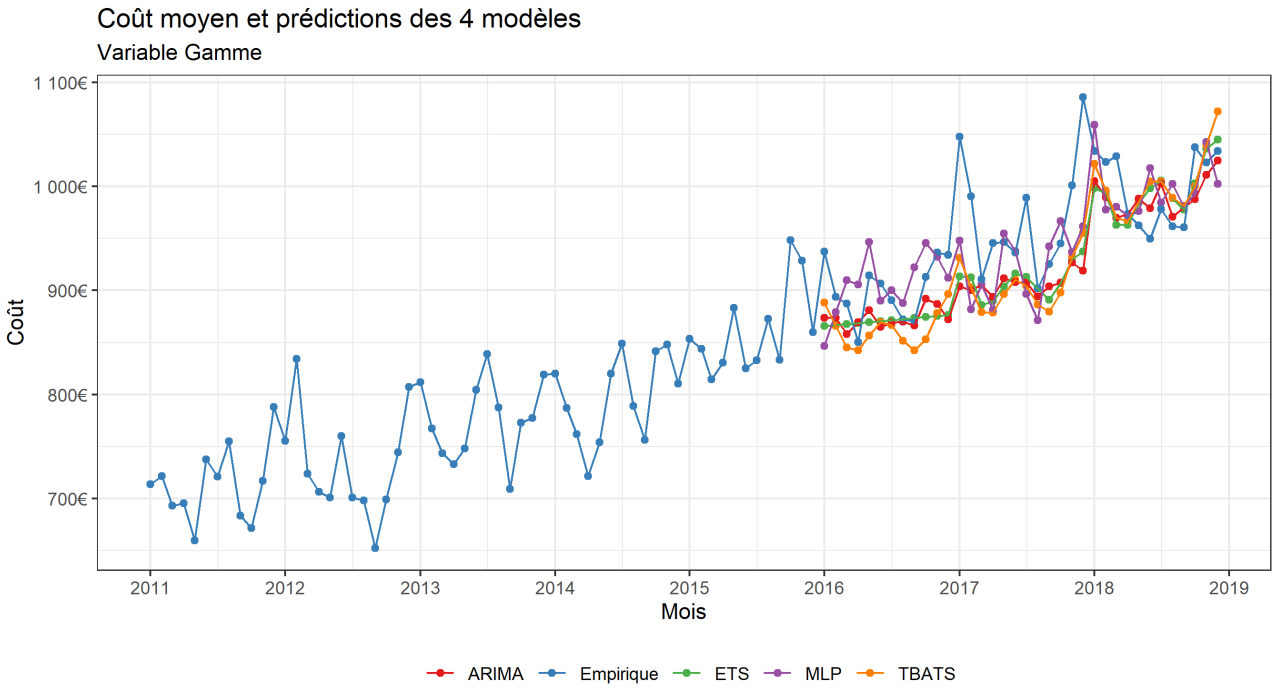
Afin d'affiner les prédictions, il s'agit d'utiliser toute l'information disponible dans nos bases de données. En effet, nous disposons, pour chaque sinistre, de variables explicatives, que nous n'utilisons pas pour l'instant.
Pour cela, nous allons diviser notre base de données suivant les modalités d'une ou plusieurs variables.
Par exemple, avec la variable Gamme, qui dispose de 4 modalités (bas de gamme, milieu de gamme, haut de gamme et autres), nous divisons la base de données suivant ces 4 modalités, puis nous appliquons le même mécanisme que pour la base de données totale, sur chacune de ces 4 bases de données.
Nous obtenons ainsi les prédictions suivantes :



Puis, afin d'effectuer la prédiction du coût moyen total nous recomposons, pour chaque mois de prédictions, le coût moyen total. Nous effectuons cette opération en effectuant la moyenne pondérée des coûts moyens de chaque gamme. Nous effectuons cette pondération avec la répartition des sinistres empiriques entre chaque gamme.
La répartition du nombre de sinistres suivant la variable "Gamme" est donné par le tableau suivant :

Gamme	**Répartition du nombre de sinistres**
Marque haute gamme	15.2%
Marque bas de gamme	58.1%
Marque moyenne gamme	16.5%
Autres	10.2%

Ainsi, en recomposant le coût moyen prédit au total, nous obtenons le graphique suivant :



De façon plus formelle, nous obtenons les écarts suivants :

Modèle	Écart absolu moyen	Écart absolu moyen en pourcentage	Erreur maximum	Erreur maximum en pourcentage
ARIMA	39.29€	4.01%	166.58€	15.3%
ETS	40.61€	4.16%	148.24€	13.7%
TBATS	43.62€	4.50%	130.53€	12.0%
MLP	38.08€	3.92%	123.96€	11.4%

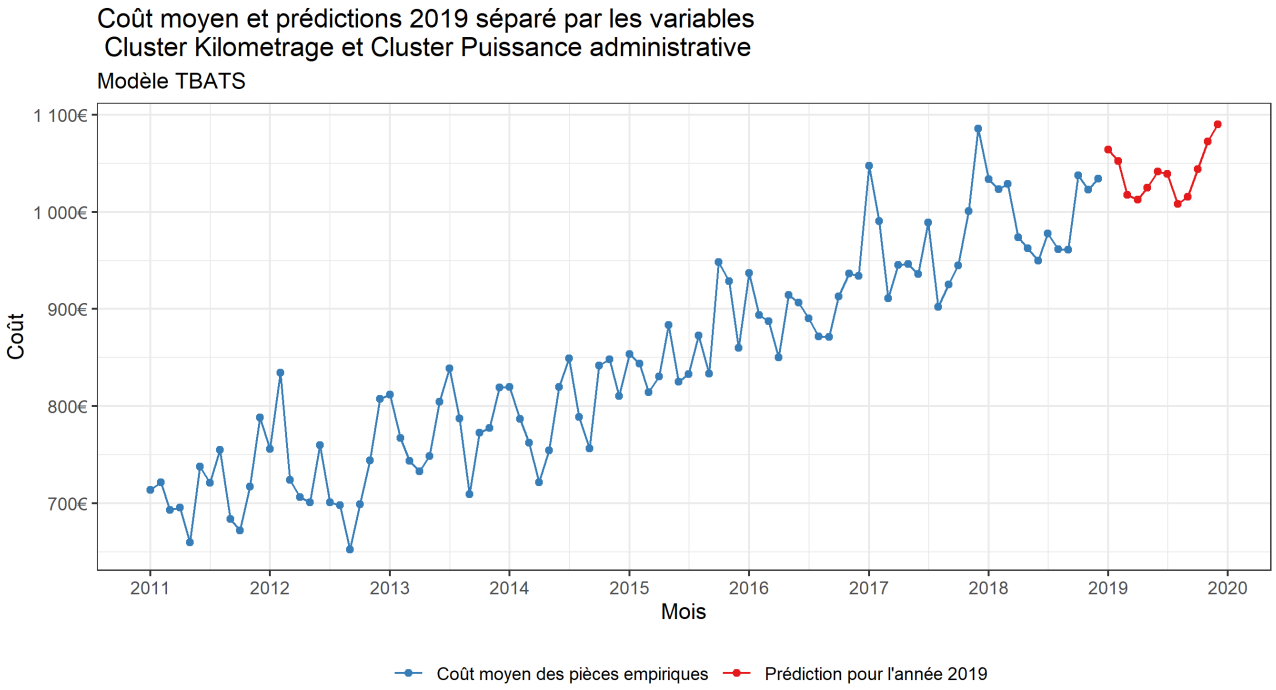
Ce tableau montre que la divisions de la base de données suivant la variable gamme n'a pas permis d'améliorer les résultats de prédiction. Nous avons donc besoin de construire des variables explicatives plus cohérentes pour notre problème. C'est ce que nous effectuerons dans la partie *Constructions de variables explicatives*, en utilisant notamment le partitionnement en k-moyennes.
De plus, nous nous autoriserons à diviser la base de données suivant plusieurs variables. Nous construirons nos 4 modèles en divisant notre base de données suivant les nouvelles variables construites, avec la même méthode que celle décrite pour la variable gamme.
Nous classons ensuite par ordre croissant les résultats suivants les valeurs de moyenne absolue des écarts (MAE).
Dans le tableau ci-dessous, RMSE représente la moyenne des écarts quadratiques à la moyenne, et ME représente l'erreur maximum.

Modele	Variable	MAE	RMSE	ME
TBATS	ClusterPUISSANCE_ADMINISTRATIVE_1-ClusterKILOMETRAGE_1	26.70810	35.79295	115.79533
ETS	GENREVEHICULE-ENERGIE_SRABIS	26.78653	36.31619	110.83804
TBATS	ENERGIE_SRABIS-ClusterKILOMETRAGE_1	27.56126	36.81385	103.80184
ETS	ENERGIE_SRABIS-ClusterKILOMETRAGE_1	27.65788	38.44097	103.47391
ETS	ENERGIE_SRABIS-ClusterPUISSANCE_ADMINISTRATIVE_1	27.77598	35.36924	97.48223
ETS	ENERGIE_SRABIS	28.07034	36.37918	95.52799
TBATS	TopAgree-ClusterPUISSANCE_ADMINISTRATIVE_1	28.43758	38.08057	118.77609
ETS	ENERGIE_SRABIS-ClusterNature_sinistre_1	28.50471	35.96410	95.85977
TBATS	Total	28.82636	37.30131	100.45526
TBATS	ClusterNB_CYLINDRE_1	29.12214	35.50489	93.87955
TBATS	ClusterCLAS_REPAR_1	29.19880	39.59384	103.40174
TBATS	ClusterAgeVehicule_2-ClusterPUISSANCE_ADMINISTRATIVE_1	29.54054	39.41961	114.11463
TBATS	GENREVEHICULE-ClusterKILOMETRAGE_1	29.60121	38.32462	106.06262
ARIMA	GENREVEHICULE-ClusterNature_sinistre_1	29.67063	41.86823	134.90161
ARIMA	ClusterPUISSANCE_ADMINISTRATIVE_1-ClusterNB_CYLINDRE_1	29.71252	41.07866	135.42117
TBATS	ClusterCLASDOM_SRA_ORIG_1	29.73404	39.15852	114.84355
ARIMA	ClusterNB_CYLINDRE_1	29.75520	40.19546	125.63306
MLP	ClusterMarque_2-ClusterCLASDOM_SRA_ORIG_1	29.83570	45.02077	147.28132
ARIMA	GENREVEHICULE-ClusterPUISSANCE_ADMINISTRATIVE_1	29.94014	42.33725	138.07615

L'analyse de ce tableau montreque 8 prédictions sont meilleures que notre prédiction de base avec le modèle TBATS.
Le meilleur modèle semble être construit avec le modèle TBATS ainsi qu'avec une séparation de la base de données suivant les deux variables "Cluster Puissance administrative" et "Cluster Kilométrage".
Pour cette prédiction, nous obtenons les résultats annuels suivants :

Année	Coût moyen empirique	Prédiction	Erreur de prédiction
2016	901.32€	895.06€	0.69%
2017	969.98€	950.44€	2.00%
2018	996.02€	1003.75€	0.78%

Nous construisons ainsi les prédictions sur l'année 2019 à l'aide de ce modèle et de cette séparation de base de données. Nous obtenons :

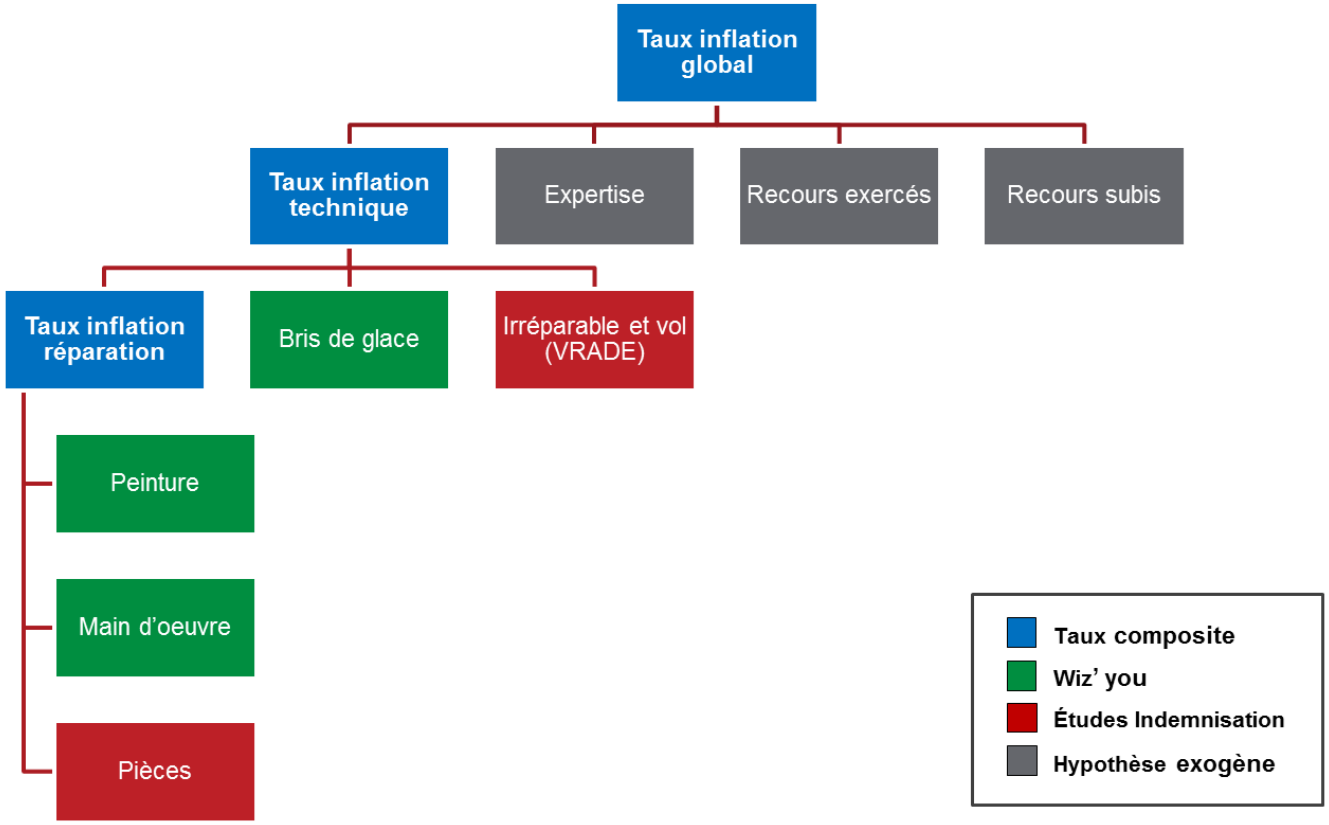


Finalement, nous établissons la prédiction finale du coût moyen des pièces pour l'année 2019, ainsi que l'inflation estimée :

Année	Coût moyen	Inflation
2011	712.85€	
2012	733.63€	2.92%
2013	777.79€	6.02%
2014	798.32€	2.64%
2015	863.13€	8.12%
2016	901.32€	4.42%
2017	969.98€	7.62%
2018	996.02€	2.68%
2019	**1040.26€**	**4.44%**

- Synthesis note -

In order to anticipate the cost of automobile claims, Generali has decided to launch the "Prospective Indicator" project to assess the evolution of costs related to automobile repairs. The study of the inflation of automobile claims costs is divided into several sections, led by two Generali teams : the "Etudes indemnisations" team and the Wiz'you team, located in Nantes. The items are distributed as follows :



In the rest of this paper, we focus on the parts item. Our work will therefore focus on studying automotive parts inflation in the context of a repair. The final objective of the brief will be to predict the average cost of parts for the year 2019

To do this, we will have a database consisting of the date of the loss, the total cost of the parts, as well as various explanatory variables.

Our study will use and compare 4 different time series models :

— The ARIMA model (Autoregressive integrated moving average)

— The ETS model (Exponential smoothing)

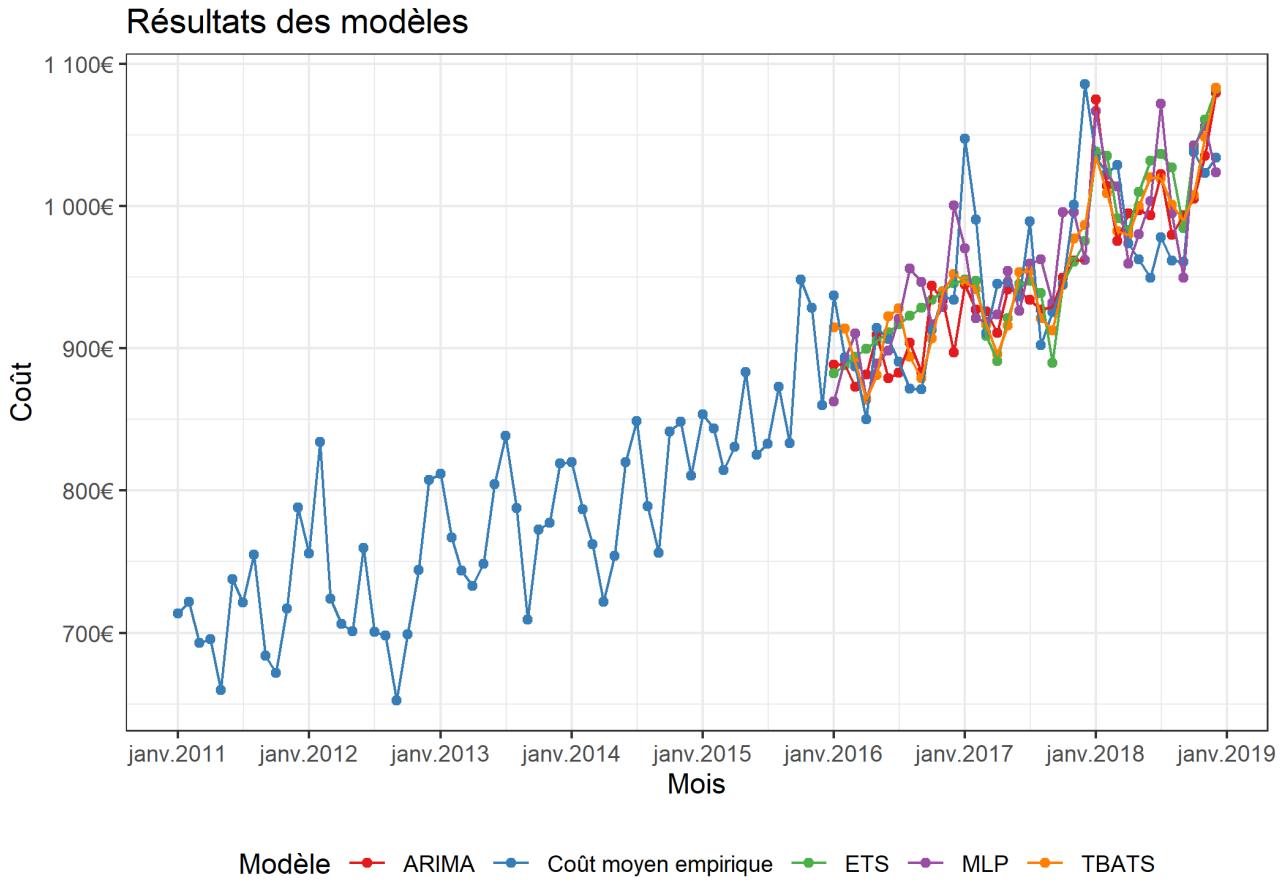
— The TBATS model (Trigonometric Seasonal, Box-Cox Transformation, ARMA residuals,Trend and Seasonality)
— The MLP model (Multi Layer Perceptron)

These models take as input a time series, i.e an ordered sequence of values, and output another time series, predicting the sequence of the input time series.
A first method for predicting inflation is :

1. Establish the average monthly cost of parts.
2. To build the different models on this test time series.
3. Establish model predictions over the 12 months following the end of the test time series
4. Predict the average annual cost by making the weighted average of the average monthly costs predicted for the year in question(this weighting is carried out with the distribution of the number of claims per month)

In order to test our models, we build the predictions on :
— The year 2016 using as a database the average cost of parts from 2011 to 2015
— The year 2017 using as a database the average cost of parts from 2011 to 2016
— The year 2018 using as a database the average cost of parts from 2011 to 2017

We get :



In order to compare the efficiency of our models, we construct, for each month of prediction and each model, the absolute error, then we average these errors. We also determine the maximum error for each model.
We get :

Model	Mean absolute errors	Mean absolute percentage error	Maximum error	Maximum percentage error
ARIMA	31.07€	3.25%	123.23€	11.4%
ETS	34.13€	3.55%	109.93€	10.1%
TBATS	28.83€	2.99%	100.46€	9.59%
MLP	33.45€	3.47%	123.41€	11.4%

We note that the TBATS model is the best over these test time series, both for absolute the mean absolute absolute deviations and for the maximum error.

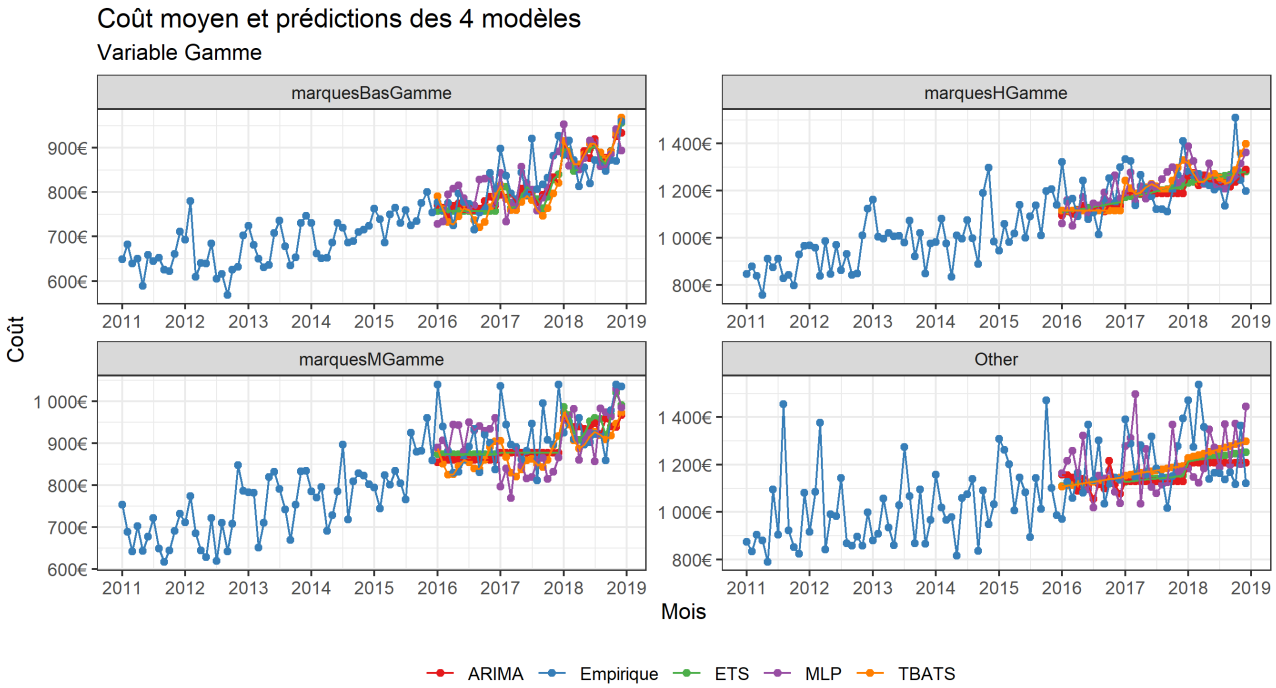
In order to predict the average cost of parts per year, we will make the weighted average of each monthly estimate. This weighting is carried out with the distribution of the number of claims per month.
The distribution of claims per month, for claims from 2011 to 2015 is :

January	February	March	April	May	June	July	August	September	October	November	December
7.94%	7.07%	7.71%	8.05%	8.50%	9.82%	8.95%	7.63%	8.61%	8.72%	8.44%	8.55%

Thus, by recomposing the average annual cost predicted using these distributions, we obtain the following results :

		Model results			
Year	Empirical average cost	ARIMA	ETS	TBATS	MLP
2016	901.32€	897.34€	914.93€	907.76€	916.02€
2017	969.98€	938.34€	935.25€	939.53€	952.23€
2018	996.02€	1013.79€	1027.15€	1015.29€	1015.32€

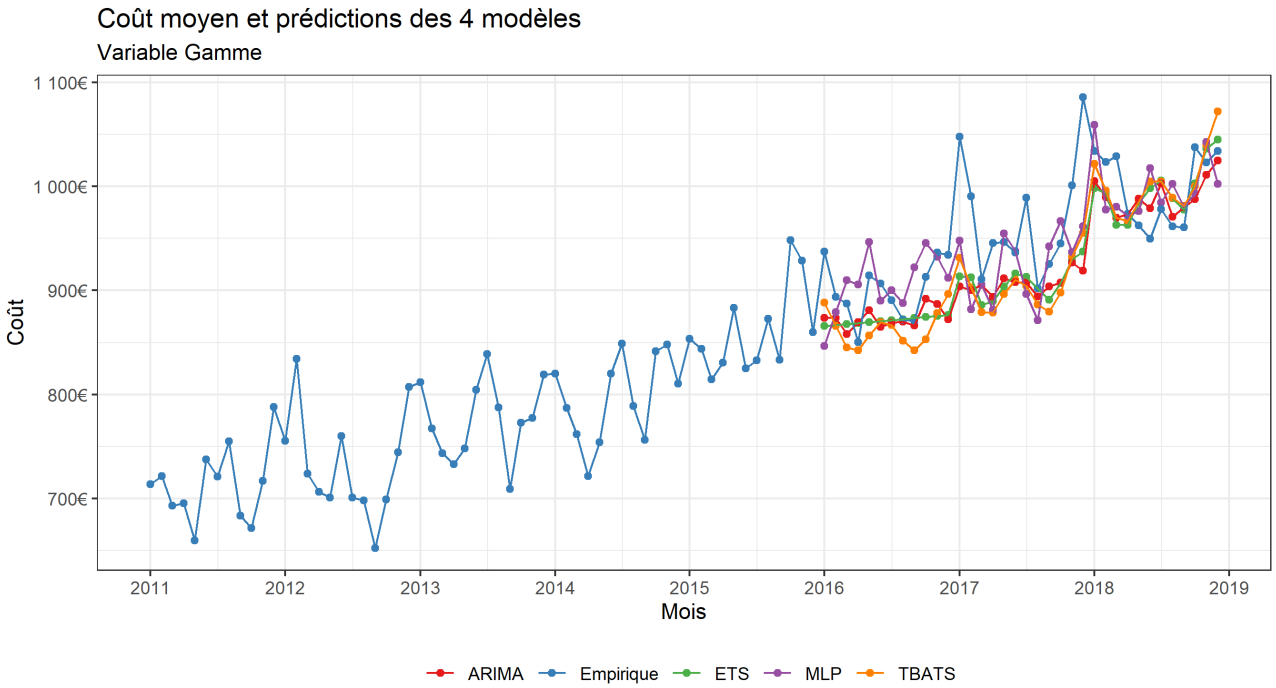
However, in order to refine the predictions, it is necessary to use all the information available in our databases. Indeed, we have explanatory variables for each claim, which we do not use at the moment.
To do this, we will divide our database according to the modalities of one or more variables. For example, with the Range variable, which has 4 modalities (low-end, mid-range, high-end, high-end, other), we divide the database according to these 4 modalities, then apply the same mechanism as for the total database, on each of the 4 databases.
We thus obtain the following predictions :



Then, in order to predict the total average cost, we recompose, for each month of predictions, the total average cost. We do this by performing the weighted average of the average costs of each Range. We carry out this weighting with the empirical distribution of claims between each range.
The distribution of the number of claims according to the "Range" variable is as follows :

Range	**Distribution of the number of claims**
High-end brand	15.2%
Low end brand	58.1%
Medium range brand	16.5%
Others	10.2%

Thus, by recomposing the average cost predicted in total, we obtain :



In a more synthetic way, we obtain the following deviations :

Modèle	Moyenne absolue des écarts	Moyenne absolue des écarts en pourcentage	Erreur maximum	Erreur maximum en pourcentage
ARIMA	39.29€	4.01%	166.58€	15.3%
ETS	40.61€	4.16%	148.24€	13.7%
TBATS	43.62€	4.50%	130.53€	12.0%
MLP	38.08€	3.92%	123.96€	11.4%

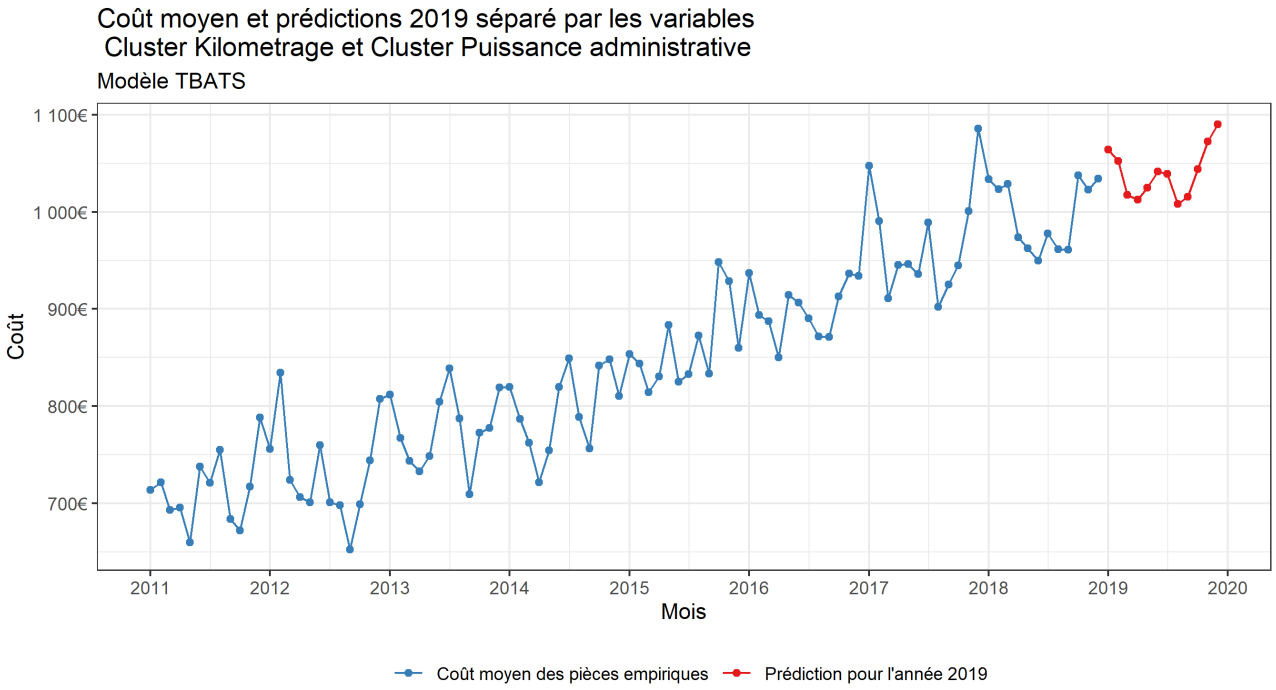
We can see that this method has not improved the results. We therefore need to build more consistent explanatory variables for our problem. This is what we do in this part, using the k-means algorithm in particular.
In addition, we will allow ourselves to divide the database according to several variables. We will build our 4 models by dividing our database according to the new variables built, using the same method as described for the range variable.
We then classify the results as follows : increasing mean absolute error (MAE). In the table below, RMSE represents the row-mean square deviation, and ME represents the maximum error.

Modele	Variable	MAE	RMSE	ME
TBATS	ClusterPUISSANCE_ADMINISTRATIVE_1-ClusterKILOMETRAGE_1	26.70810	35.79295	115.79533
ETS	GENREVEHICULE-ENERGIE_SRABIS	26.78653	36.31619	110.83804
TBATS	ENERGIE_SRABIS-ClusterKILOMETRAGE_1	27.56126	36.81385	103.80184
ETS	ENERGIE_SRABIS-ClusterKILOMETRAGE_1	27.65788	38.44097	103.47391
ETS	ENERGIE_SRABIS-ClusterPUISSANCE_ADMINISTRATIVE_1	27.77598	35.36924	97.48223
ETS	ENERGIE_SRABIS	28.07034	36.37918	95.52799
TBATS	TopAgree-ClusterPUISSANCE_ADMINISTRATIVE_1	28.43758	38.08057	118.77609
ETS	ENERGIE_SRABIS-ClusterNature_sinistre_1	28.50471	35.96410	95.85977
TBATS	Total	28.82636	37.30131	100.45526
TBATS	ClusterNB_CYLINDRE_1	29.12214	35.50489	93.87955
TBATS	ClusterCLAS_REPAR_1	29.19880	39.59384	103.40174
TBATS	ClusterAgeVehicule_2-ClusterPUISSANCE_ADMINISTRATIVE_1	29.54054	39.41961	114.11463
TBATS	GENREVEHICULE-ClusterKILOMETRAGE_1	29.60121	38.32462	106.06262
ARIMA	GENREVEHICULE-ClusterNature_sinistre_1	29.67063	41.86823	134.90161
ARIMA	ClusterPUISSANCE_ADMINISTRATIVE_1-ClusterNB_CYLINDRE_1	29.71252	41.07866	135.42117
TBATS	ClusterCLASDOM_SRA_ORIG_1	29.73404	39.15852	114.84355
ARIMA	ClusterNB_CYLINDRE_1	29.75520	40.19546	125.63306
MLP	ClusterMarque_2-ClusterCLASDOM_SRA_ORIG_1	29.83570	45.02077	147.28132
ARIMA	GENREVEHICULE-ClusterPUISSANCE_ADMINISTRATIVE_1	29.94014	42.33725	138.07615

We observe that 8 predictions are better than our basic prediction with the TBATS model. The best model seems to be built with the TBATS model as well as with a separation of the database according to the two variables "Cluster Administrative Power" and "Cluster Mileage". For this prediction, we obtain the following annual results :

Year	Empirical average cost	Prediction	Prediction error
2016	901.32€	895.06€	0.69%
2017	969.98€	950.44€	2.00%
2018	996.02€	1003.75€	0.78%

We are thus constructing predictions for the year 2019 using this model and database separation. We get :



Finally, we establish the final prediction of the average cost of parts for the year 2019, as well as the estimated inflation :

Year	Average cost	Inflation
2011	712.85€	
2012	733.63€	2.92%
2013	777.79€	6.02%
2014	798.32€	2.64%
2015	863.13€	8.12%
2016	901.32€	4.42%
2017	969.98€	7.62%
2018	996.02€	2.68%
2019	**1040.26€**	**4.44%**

Table des matières

Chapitre 1

Contexte

Le secteur de l'assurance automobile est en constante évolution, notamment de part les différentes évolutions technologiques qui rendent la voiture d'aujourd'hui et de demain plus sûre. Cela se traduit par une diminution du nombre de sinistres mais un augmentation des coûts de réparation. En effet, l'inflation observée dans le domaine de la réparation automobile est bien supérieure au taux d'inflation français. C'est pourquoi, une étude approfondie de cette inflation est indispensable pour une compagnie d'assurance,notamment pour les revalorisations tarifaires annuelles notamment. De plus l'augmentation du nombre de données disponibles ainsi que l'émergence de nouveaux modèles rendant les prédictions de plus en plus précises, permettent l'élaboration de nouveaux indicateurs, ainsi qu'un suivi plus précis de la sinistralité.

Nous allons ainsi développer, tout au long de ce mémoire, des méthodes afin de déterminer le coût moyen futur du poste pièces lors de la réparation d'un véhicule dans le cadre d'un sinistre automobile matériel.

Cependant, ces méthodes développés pourront se généraliser à l'étude de tout coût moyen, à condition de disposer d'un nombre suffisant de données comme nous le verrons ci-dessous.

1.1 Présentation de Generali et du service Études Indemnisation

1.1.1 Generali

Fondées en 1831 en Italie, les « Assicurazioni Generali Austro-Italiche » ont donné naissance aux premières assurances généralistes, dont la Compagnie,Generali doit d'ailleurs son nom à cette caractéristique. En effet, elle est la première à avoir proposé à ses clients une couverture multirisques.

Le groupe Generali est aujourd'hui l'un des principaux groupes mondiaux d'assurance et de services financiers. Troisième au rang mondial derrière Allianz et AXA, le groupe est présent et opère dans plus de 60 pays desservant au moins 55 millions de clients, et s'appuie sur le savoir-faire de plus de 70 000 collaborateurs à travers le monde.

Un an après sa création en Italie, Generali ouvre sa première agence en France en 1832, symbole d'une expansion internationale de la compagnie, puisqu'il s'agit de sa première implantation en dehors de ses frontières. Très vite, la filiale du Groupe acquiert progressivement diverses sociétés de l'Hexagone pour aboutir à la création de Generali France en 1995, qui devient une référence majeure sur le marché. La France est actuellement le troisième marché du groupe, après l'Italie et l'Allemagne.

Generali s'engage à l'anticipation et à l'atténuation des risques de ses clients. De ce fait, la Compagnie propose une gamme de produits et services qui couvrent tous les besoins : les assurances de personnes (Generali Vie), les assurance de biens et de responsabilité (Generali IARD) et l'assistance. Ainsi, elle joue un rôle pro actif et déterminant dans l'amélioration par

l'assurance, de la vie de tous.

1.1.2 Le service Études Indemnisation

Cette direction,intègre au sein de la TA Non-vie (Techniques actuarielles non vie) rassemble les services techniques et actuariels de l'assurance non-vie, repartis selon les branches dommages (Automobile, Dommages aux Biens, Responsabilité Civile,...). Dirigé par Jean-Sébastien VIEU, le service Études Indemnisation est un service transverse d'études techniques.

Le service Études Indemnisation agit comme une « tour de contrôle », capable de produire un suivi technique permettant de suivre la sinistralité des branches non-vie, par le biais d'indicateurs suffisamment pertinents pour pouvoir en donner une situation de leur état de santé. Toutes ces études passent par les analyses du nombre de sinistres enregistrés, les coûts moyens ou encore la robustesse des provisions, pour ne citer qu'eux.

En plus du pilotage de la sinistralité, ce service participe à des études statistiques ponctuelles, notamment pour l'aide au provisionnement des branches à développement long (comme pour la branche Construction, par exemple). Le service s'adonne aussi à des exercices de prédictions du nombre et de la charge des sinistres, en cas d'évènements naturels par exemple. C'est un service transverse qui travaille en collaboration avec plusieurs autres départements : le pôle Risques Naturels, la Réassurance, le Provisionnement, et la Finance, entre autres.

C'est dans ce cadre qu'est né le projet "Indicateur prospectif".

1.2 Indicateur prospectif : Le projet

Débuté en août 2018, le projet Indicateur prospectif est mené conjointement par deux équipes :

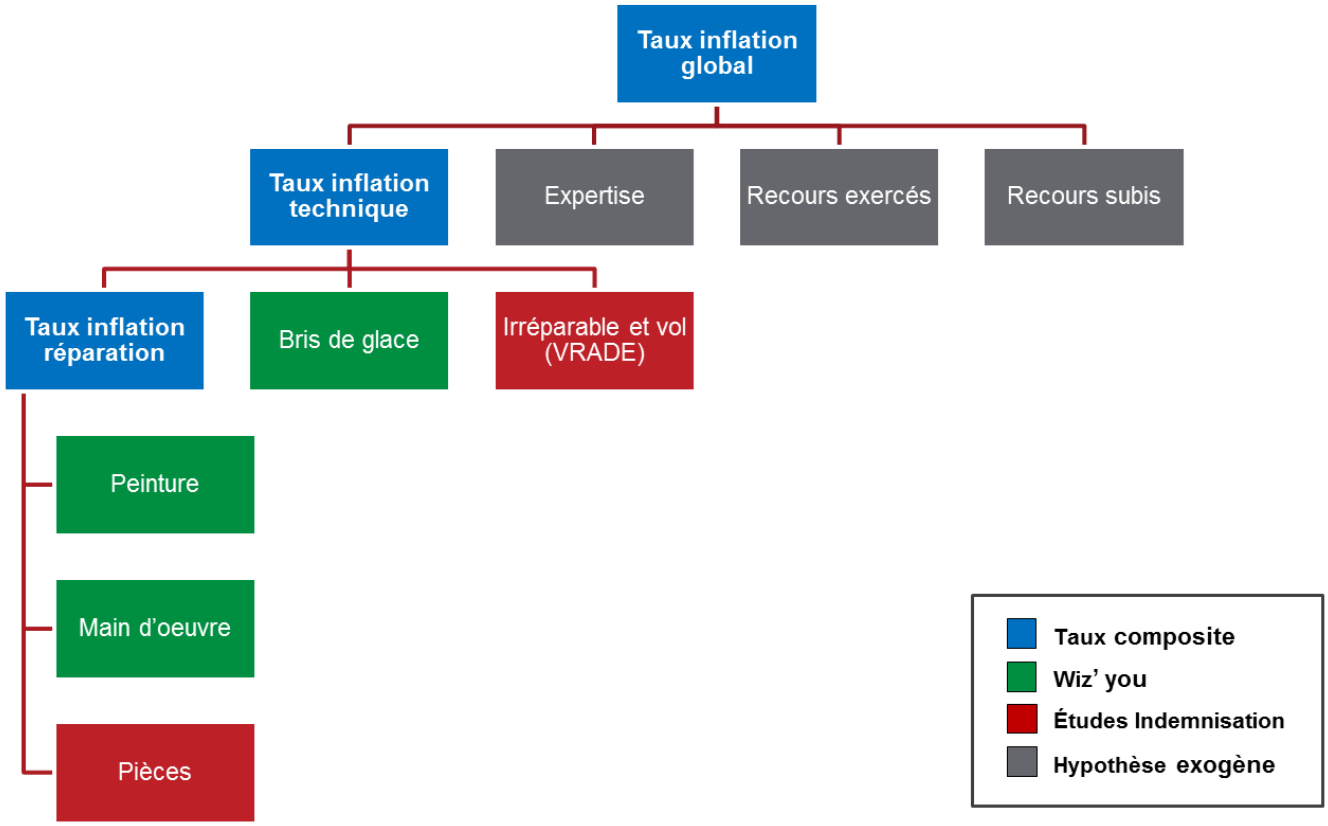
- L'équipe Études Indemnisation
- L'équipe Wiz'you, localisé à Nantes et responsables de diverses initiatives sur l'indemnisation

Ce projet a deux buts principaux :

- Évaluer l'évolution des coûts liés à la réparation des véhicules afin d'estimer le coût moyen Auto à 1 an (prévisions budgétaires)
- Mettre en place des indicateurs de suivi des coûts de réparation propres à Generali (En comparaison avec les indicateurs SRA, propres au marché automobile en général).

L'étude du coût de réparation des véhicules a été divisé en plusieurs branches, réparties entre les deux équipes.

Elle se divise comme ceci :



On se concentrera, dans la suite de ce mémoire, sur la partie "Pièces". L'objectif sera ainsi de déterminer l'inflation moyenne sur 2019 du coût des pièces.
Cependant, les méthodes développées dans ce mémoire sont applicables aux autres branches du projet.

1.3 Objectif du mémoire et principes clés

L'objectif de ce mémoire sera ainsi de développer plusieurs modèles afin d'estimer le coût moyen des pièces pour l'année N+1.

Nous réaliserons la conception de modèles de séries temporelles, permettant, avec des données de 2011 à 2018, de prédire le coût moyen des pièces en 2019.
Pour cela, nous aurons à notre disposition une base de données contenant, pour chaque sinistre automobile, le coût des pièces pour le sinistre, ainsi que différentes informations sur le sinistre et le contrat. Nous détaillerons la construction de cette base de données dans la partie *Construction de la base de données*.

Pour déterminer l'inflation du coût moyen des pièces pour l'année N+1, nous allons déterminer ce coût pour chaque mois de l'année N+1, pour, ensuite recombiner le coût moyen des pièces pour l'année en question en faisant une somme pondérée des coûts moyens de chaque

mois. Pour matérialiser ceci mathématiquement, introduisons des notations :

Notons
— y_i le coût moyen des pièces pour le mois i
— n_i le nombre de sinistres pour le mois i
— c_i le coût total des pièces pour le mois i
— Y_n le coût moyen des pièces pour l'année n
— N_n le nombre de sinistres pour l'année n
— C_n le coût total des pièces pour l'année n

Nous avons ainsi,

— $\forall i \in \mathbb{N} \quad y_i = \frac{n_i}{c_i}$
— $\forall n \in \mathbb{N} \quad Y_n = \frac{N_n}{C_n}$

Avec notre base de données, nous connaissons ainsi
— Y_n, N_n, C_n pour $n \in [\![2011, 2018]\!]$
— y_i, n_i, c_i pour $i \in [\![1, 96]\!]$

Notre but est ainsi d'estimer Y_{2019}

Pour cela nous avons deux méthodes intuitives qui s'offrent à nous :

Méthode Charge/Nombre
$\forall i \in [\![97, 119]\!]$ nous estimons n_i et c_i, puis nous construisons :

$$\hat{N}_{2019} = \sum_{i=97}^{119} \hat{n}_i \quad et \quad \hat{C}_{2019} = \sum_{i=97}^{119} \hat{c}_i \tag{1.1}$$

puis finalement nous construisons

$$\hat{Y}_{2019} = \frac{\hat{C}_{2019}}{\hat{N}_{2019}} \tag{1.2}$$

Méthode coût moyen pondéré
$\forall i \in [\![97, 119]\!]$ nous estimons y_i. Puis, nous calculons la répartition du nombre de sinistres pour le mois i. Pour cela, on prend simplement la répartition empirique du nombre de sinistres pour le mois en question. Nous avons ainsi : $\forall i \in [\![97, 119]\!]$

$$\hat{r}_i = \frac{\sum_{k=1}^{8} n_{i-k}}{\sum_{k=1}^{96} n_k} \tag{1.3}$$

Nous reconstituons ensuite $\hat{Y}_{2019}$ avec :

$$\hat{Y}_{2019} = \sum_{i=97}^{119} \hat{y}_i \times \hat{r}_i$$

On se concentrera, dans la suite de ce mémoire, sur la deuxième méthode. En effet la première méthode donne de moins bons résultats, ce qui est dû à la forte volatilité du nombre de sinistres et du coût total.

Notre but est maintenant d'obtenir la meilleure estimation de y_i

.
Pour cela, une première approche est d'utiliser les $y_1, y_2, ...y_{96}$ pour estimer les $y_{97}, y_{98}, ...y_{119}$.

Cependant, en agrégeant les coûts des pièces par mois, on perd toute l'information de nos variables explicatives, comme par exemple la marque du véhicule.

On va donc plutôt diviser notre base de données d'entraînement suivant certaines modalités de variables. Par exemple, pour la variable *Gamme du véhicule*, on va diviser notre base d'entraînement en 4 plus petites bases :

— Une base constituée des sinistres des véhicules "Bas de gamme"
— Une autre pour ceux de "Moyenne gamme"
— Une autre pour ceux de "Haut de gamme
— Une autre pour les véhicules restants

On estimera ensuite le coût moyen pour chaque mois sur chacune de ces bases d'entraînements, puis on reconstituera le coût moyen total des pièces. Pour recomposer le coût moyen, nous pondérons les coûts moyens pour chaque base d'entraînement avec la répartition empirique des sinistres pour chaque catégorie.

On introduit, de façon plus formelle :

— y_i^{BG} le coût moyen des pièces pour la base de données "Bas de gamme"
— y_i^{MG} le coût moyen des pièces pour la base de données "Moyenne gamme"
— y_i^{HG} le coût moyen des pièces pour la base de données "Haut de gamme"
— y_i^{A} le coût moyen des pièces pour la base de données "Autres"
.

On construit aussi la répartition empirique des sinistres pour chaque catégorie.

— $\hat{r}^{BG} = \frac{\sum_{k=1}^{96} n_k^{BG}}{\sum_{k=1}^{96} n_k}$
— $\hat{r}^{MG} = \frac{\sum_{k=1}^{96} n_k^{MG}}{\sum_{k=1}^{96} n_k}$
— $\hat{r}^{HG} = \frac{\sum_{k=1}^{96} n_k^{HG}}{\sum_{k=1}^{96} n_k}$
— $\hat{r}^{A} = \frac{\sum_{k=1}^{96} n_k^{A}}{\sum_{k=1}^{96} n_k}$

On reconstruit ainsi l'estimation totale de y_i avec :

$$\hat{y}_i = \hat{y}_i^{BG} \times \hat{r}^{BG} + \hat{y}_i^{MG} \times \hat{r}^{MG} + \hat{y}_i^{HG} \times \hat{r}^{HG} + \hat{y}_i^{A} \times \hat{r}^{A} \tag{1.4}$$

On peut aussi diviser la base de données suivant plusieurs variables. Par exemple en divisant la base de donnée suivant la gamme et le genre de véhicule on diviserait la base de donnée en 8 bases plus petites. :

— Bas de gamme - Véhicule particulier
— Bas de gamme - Camionnette
— Moyenne gamme - Véhicule particulier

— ...

— Autres - Camionnette

Cependant, il s'agira de ne pas diviser la base de données en un nombre trop conséquent de plus petites bases.
En effet, dans ce cas, chaque base serait alors constituée d'un faible nombre d'observations, et donc en agrégeant les coûts moyens de pièces sur chacune de ces bases, la notion de coût moyen ne serait plus pertinente.

Ainsi, une des problématiques de ce mémoire sera aussi d'agréger certaines variables en un nombre plus faible de modalités, modalités qui devront être cohérentes pour prédire le coût moyen des pièces .
Par exemple dans notre exemple ci dessus, on a résumé la variable *MARQUE*, contenant initialement 197 marques différentes en la variable *GAMME* contenant 4 modalités.
Bien sûr, nous chercherons à optimiser cette agrégation pour fournir la meilleure estimation du coût moyen des pièces. Nous verrons comment procéder dans la partie *Construction de la base de données*.

Chapitre 2

Travail sur la qualité des données et détection des valeurs aberrantes

Pour construire notre base de données, nous nous servirons de plusieurs sources de données, que nous détaillerons ci dessous.

2.1 Présentation des données

2.1.1 Base de donnée DARVA

Présentation de DARVA

Pour chaque sinistre automobile, un rapport d'expertise détaillant le coût du sinistre poste par poste, pièce par pièces est émis. Generali a accès à ces données depuis l'année 2011 via DARVA.

DARVA (Développement d'applications sur réseaux à valeur ajoutée)est une société spécialisée en échange de données informatisées(EDI) et solutions Internet pour les métiers de l'assurance. Sa vocation est de contribuer à la simplification des relations entre les systèmes informatiques des assureurs et ceux de leurs partenaires dans le cadre des échanges de données pour l'assurance automobile, habitation et santé.
Ainsi depuis 2011, DARVA fournit à Generali tous les rapports d'expertises des sinistres automobiles (excepté pour les sinistres bris de glaces).

Détail de la base de données DARVA

La base fournit ainsi, pour chaque sinistre, la liste des pièces remplacées, leur coût hors taxe et taxe incluse ainsi que diverses informations sur le contrat et le véhicule endommagé.
Dans un premier temps, nous agrégeons cette base selon le numéro de sinistre, nous sommons le coût de toutes les pièces utilisées lors de la réparation. Nous constituons ainsi une base de données de **430240 lignes** et de **11 colonnes**. Nos 11 colonnes sont identifies comme suit :

- La référence du sinistre (i.e le numéro de sinistre)
- le code postal du réparateur
- la date de première immatriculation
- la date du sinistre
- le genre du véhicule
- la marque du véhicule
- Le top agrée (Désigne si le réparateur appartient au réseau de garages agrée)
- Le kilométrage du véhicule au moment du sinistre
- le modèle du véhicule
- le numéro de série (Numéro permettant d'identifier un véhicule)
- le prix total de la mission

2.1.2 Jointure avec Moninpr

Afin d'obtenir de nouvelles variables explicatives, nous effectuons une jointure à gauche de la base de données DARVA avec la base de données classique des sinistres Generali, appelée Moninpr. Nous effectuons ainsi une jointure par le numéro de sinistre.
Comme nous effectuons une jointure à gauche avec la base de données DARVA, notre but est de trouver le plus de sinistres appartenant à la base DARVA dans la base Moninpr. Ainsi, nous extrayons une base de données sur tous les périmètre de l'assurance Auto/mono, bris de glace inclus.
Notre base de données Moninpr est ainsi constituée de **2053785 lignes** et de **7 colonnes**. Nos 7 colonnes sont détaillés ci-dessous :

- Le numéro de sinistre
- Le coût du sinistre avec forfait d'expertise
- Le coût du sinistre sans forfait d'expertise
- La date de survenance du sinistre
- La nature du sinistre
- Le regroupement produit
- La responsabilité du sinistre

Après cette opération de jointure, nous retrouvons **418017 lignes** soit un taux de sinistres non retrouvé de **2.84%**.
Notre base de données est ensuite filtré pour prendre uniquement les sinistres Auto/Mono (on exclut les sinistres flottes et garages, non utiles pour notre étude).
On obtient ainsi une base de données avec **323510 lignes.**

2.1.3 Jointure avec le véhiculier SRA

Présentation de SRA

SRA (SÉCURITÉ et RÉPARATION AUTOMOBILES) est un organisme professionnel, créé en 1977. Il a le statut d'association loi 1901. Toutes les entreprises d'assurances automobiles sont adhérentes.

Ainsi, SRA fournit à Generali, chaque semaine, un véhiculier, détaillant pour chaque véhicule (repéré par le numéro de série) toutes les caractéristiques du véhicule ainsi que certains regroupements que nous détaillons ci dessous

Détail des variables du véhiculier retenues

Nous retenons ainsi **9 colonnes** :

- Le code APSAD, ie le numéro de série du véhicule
- La classe de réparation SRA
- Le groupe SRA d'origine
- la classe dommage SRA d'origine
- la carrosserie
- le type de carburant
- la puissance administrative
- Le nombre de cylindres du véhicule
- La vitesse maximum du véhicule

Après la jointure , nous arrivons à retrouver ces variables pour **98.25%** des sinistres. Pour les **1.25%** restants , nous affectons la valeur "NAN" aux lignes concernées.

2.1.4 Jointure avec le zonier Generali

Afin d'utiliser au mieux l'information du code postal du réparateur, nous effectuons une jointure de notre base de données avec le zonier de Generali.
Ce zonier fournit pour chaque commune française repérée par son code INSEE, une "note" de 1 à 20 reflétant le niveau de risque de la commune en question.

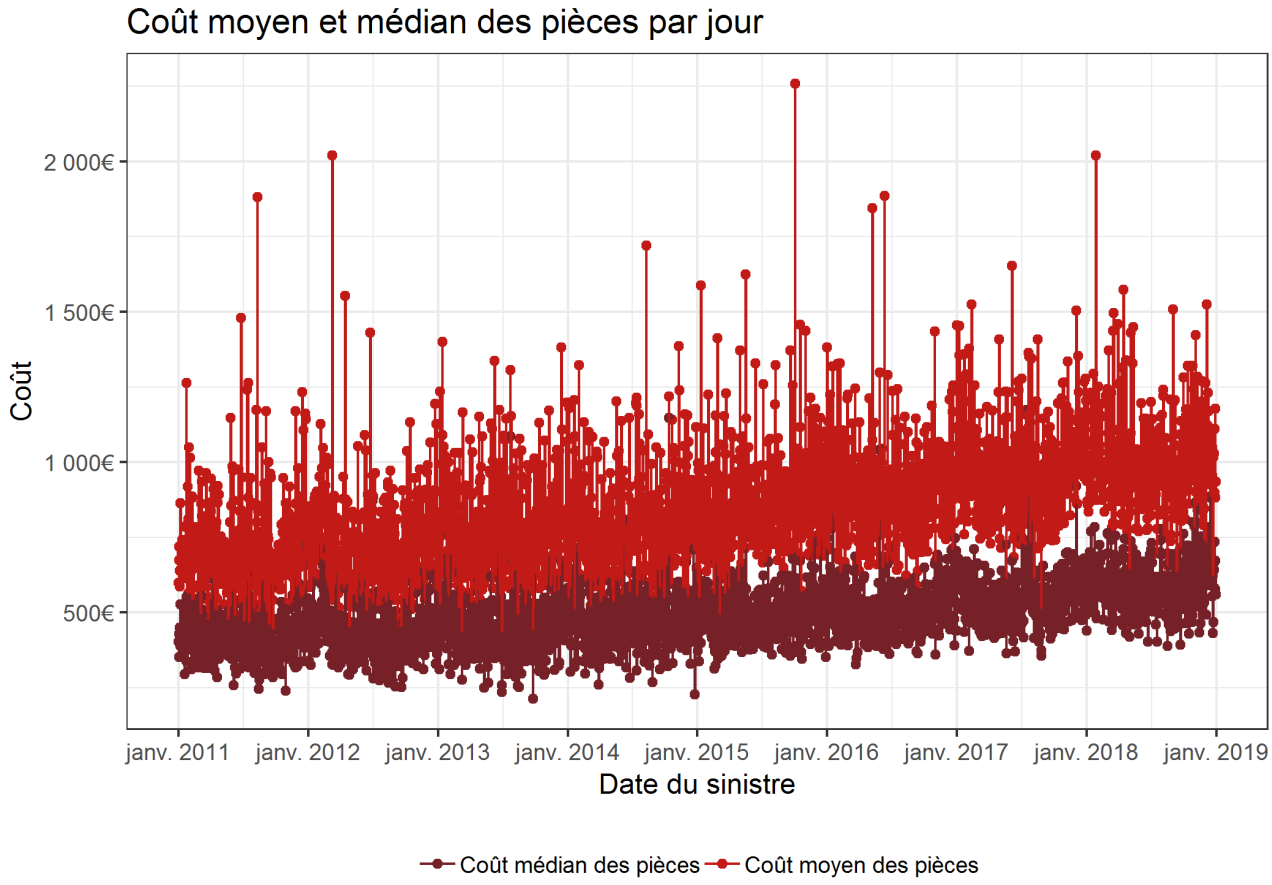
Cependant, notre base de données nous donne uniquement le code postal du réparateur, quand le zonier est fourni par code INSEE.
Nous reconstituons ainsi pour chaque code postal de France une "note" de 1 à 20 comme étant la moyenne des notes de chacune des communes du code postal en question. Nous effectuons ensuite la jointure avec notre base de données.
Nous arrivons finalement à retrouver la note de notre zonier pour **96.8%** des sinistres.
Ainsi pour les **3.24%** de sinistres restants, nous affectons la valeur "NAN" à la variable "ZoneDOM".

2.2 Suppression des valeurs aberrantes

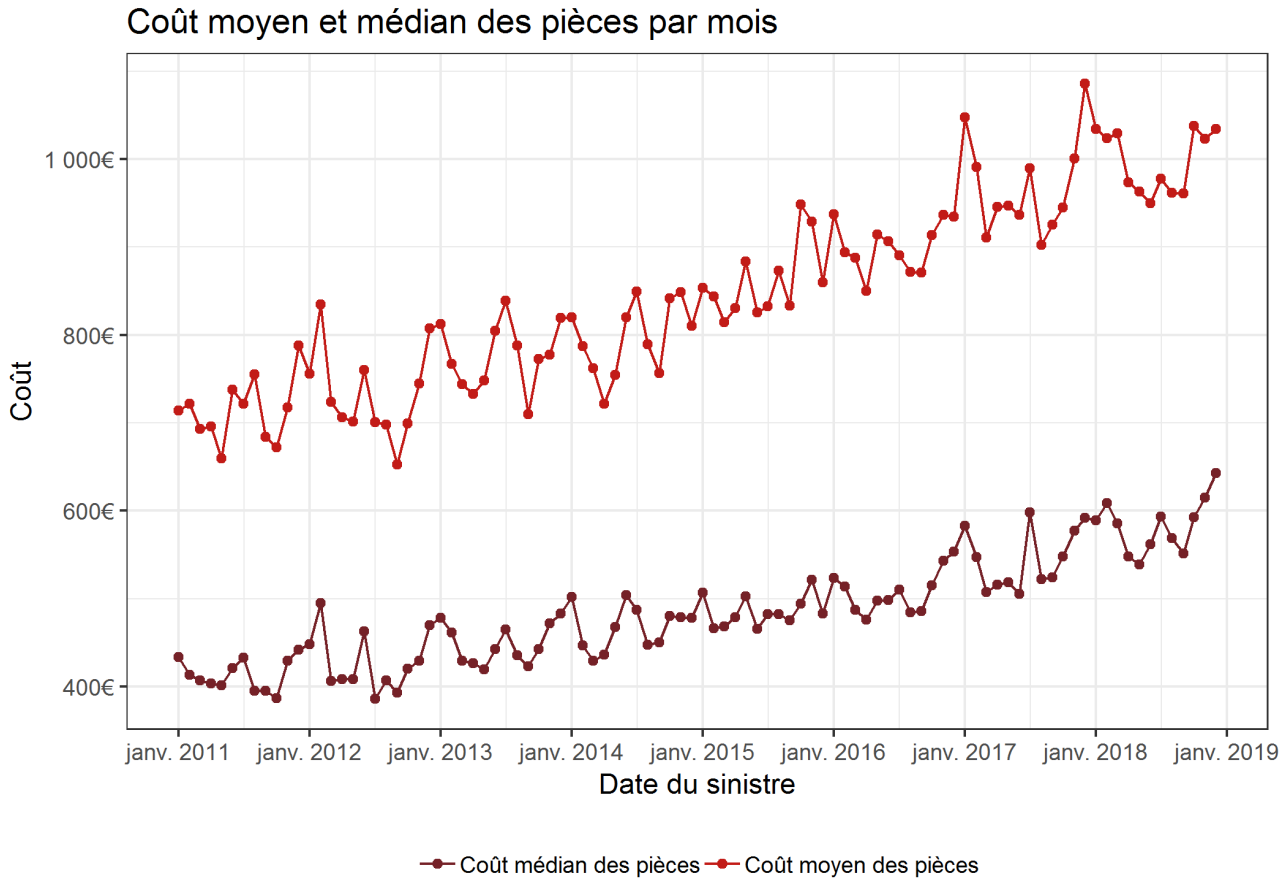
Les rapports d'expertises DARVA sont, en premier lieu, remplis à la main avant d'être rentrés dans le système informatique.
C'est pourquoi des erreurs subsistent dans la base de données, et certains montants sont erronés. Il s'agit alors de détecter et de supprimer ceux qui auront le plus de poids dans nos modèles.
Pour cela, nous vérifions manuellement si les montants reportés dans la base de données DARVA sont bien conformes à ceux mentionnés dans le système de gestion interne de Generali, ULIS.
Nous nous limitons au contrôle des 200 montants de sinistres les plus élevés présents dans notre base de données.
Parmi ces 200 sinistres, nous supprimons **73** sinistres de notre base de données.

2.3 Statistiques descriptives de la base de données

Notre base de données finale comprend **323383 lignes**.
Les coûts moyens des pièces et coûts médians des pièces par jour sont les suivants :



On observe une forte volatilité de nos coûts journalier, comme en témoigne la différence entre coût moyen et coût médian. En effet, un coût moyen haut par rapport au coût médian implique la présence d'outliers dans notre base de données.
Afin de vérifier cela, on trace le coût moyen et le coût médian par mois.



On observe ici une forte volatilité du coût moyen mensuels, notamment pour l'année 2017. Ceci pourra perturber l'entraînement de nos modèles, mais nous y reviendrons ultérieurement.
En regardant les coûts par année, on obtient les chiffres suivants :

Année	Coût moyen des pièces	Inflation annuelle du coût moyen
2011	712.85€	
2012	733.63€	2.92%
2013	777.79€	6.02%
2014	798.32€	2.64%
2015	863.13€	8.12%
2016	901.32€	4.42%
2017	969.98€	7.62%
2018	996.02€	2.68%

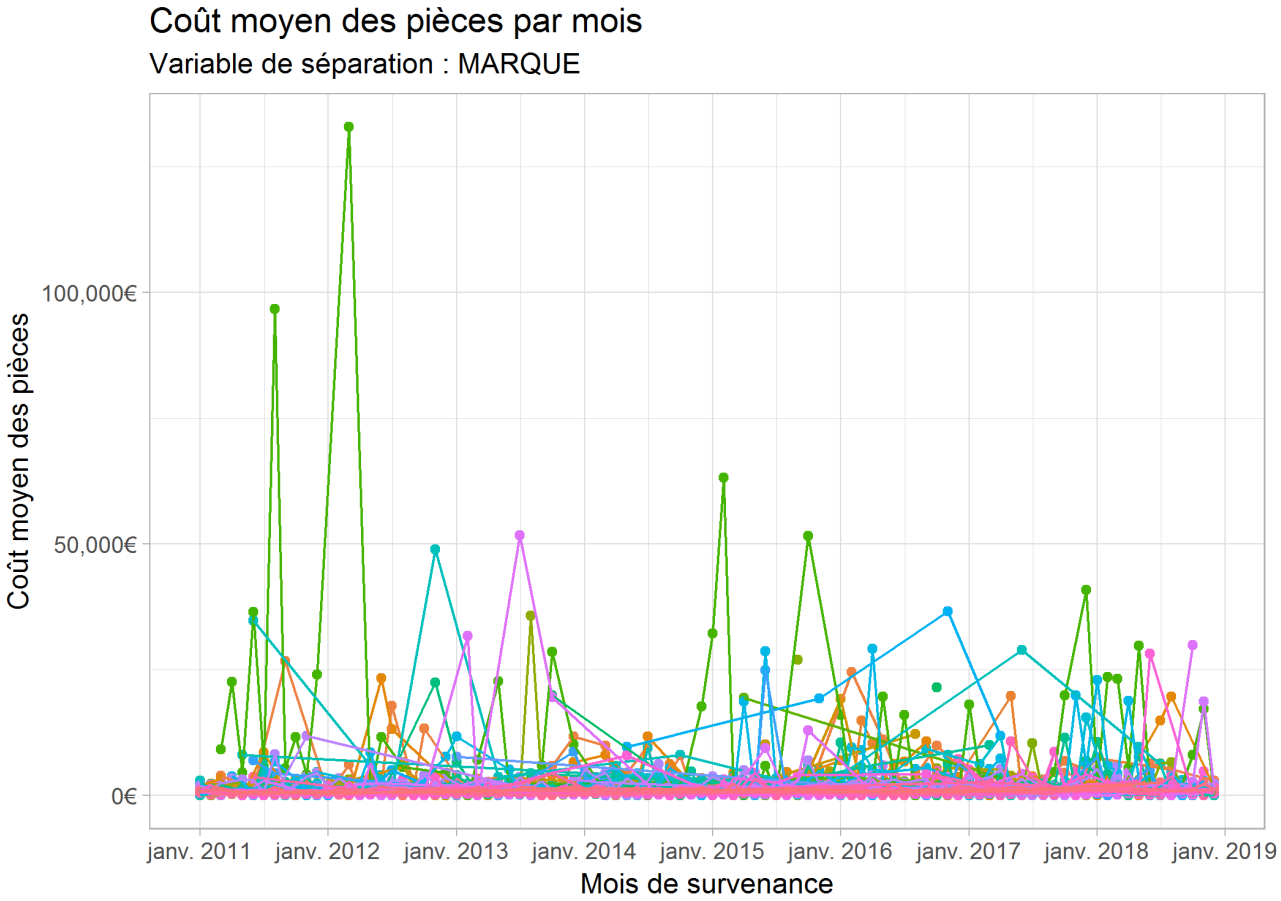
Année	Coût médian des pièces	Inflation annuelle du coût médian
2011	412.48€	
2012	428.95€	3.99%
2013	447.97€	4.43%
2014	466.62€	4.16%
2015	485.25€	3.99%
2016	507.47€	4.58%
2017	545.81€	7.56%
2018	581.34€	6.51%

On remarque ainsi que l'inflation du coût moyen des pièces oscille entre 2,68% et 8,12%, sans motif visible. L'inflation du coût médian des pièces est quant à elle, plus stable. Cela nous confirme la forte volatilité de nos données, et donc de nos coûts moyens. Il s'agira de garder cela en tête au moment de juger l'efficacité de nos différents modèles.

Chapitre 3

Création des variables explicatives

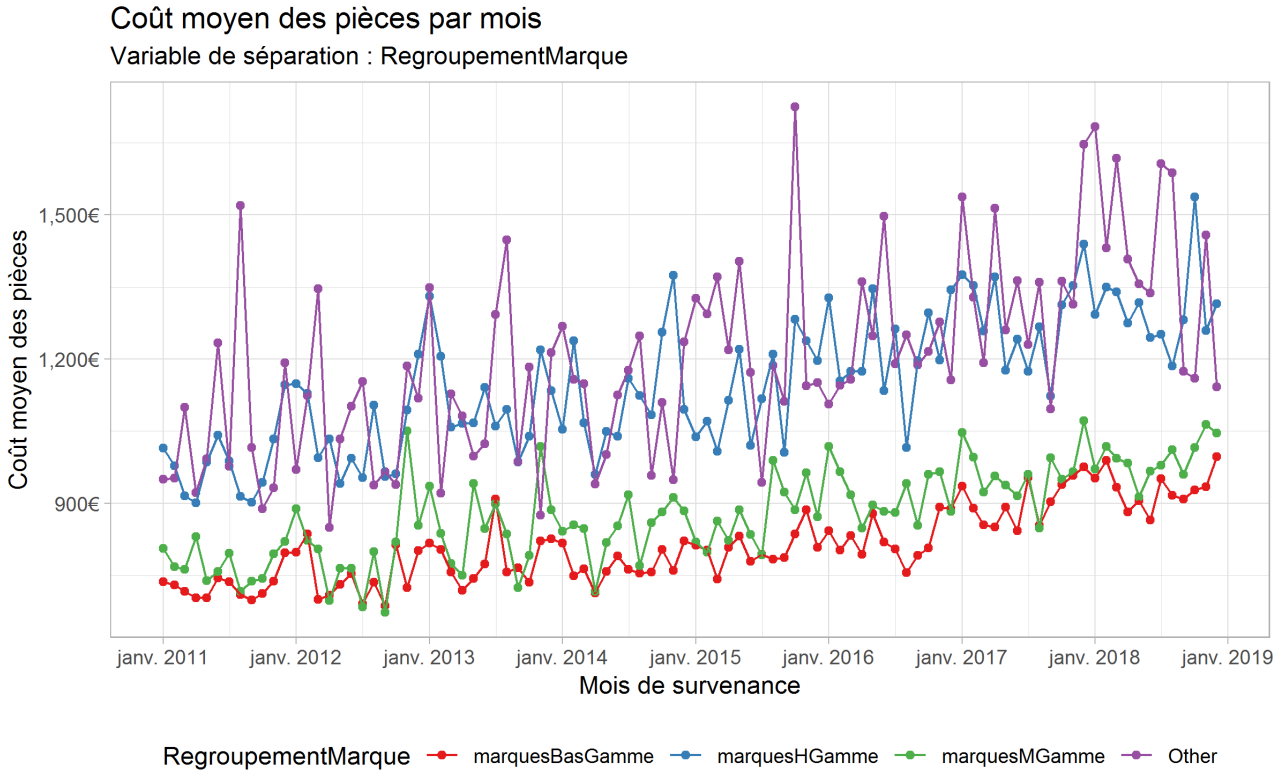
Après avoir effectué nos différentes jointures, nous possédons beaucoup de variables explicatives. Cependant, celles ci ont trop de modalités pour notre problématique.
Par exemple, voici le coût moyen des pièces observés si on sépare la base de données par marque (composé de 96 modalités) :



Nous voyons bien ici qu'un faible nombre de marques, très onéreuses à réparer, tirent les coûts moyens vers le haut et rendent leur lecture difficile.
Nous avons donc besoin de résumer ces variables en des classes plus petites.
Dans le cadre du projet "Indicateur prospectif", les regroupements de ces variables sont réalisés "à dire d'expert".
Dans notre cas, la variable *MARQUE* est résumé dans la variable *GAMME* comprenant les 4 modalités suivantes :

— "Haut de gamme" incluant les marques BMW, Audi, Mercedes, Mini
— "Milieu de gamme incluant les marques Fiat, Nissan, Toyota, Volkswagen,
— "Bas de gamme" incluant les marques Seat, Opel, Ford, Dacia, Citroën, Peugeot, Renault
— "Autres" incluant toute les autres marques

Ainsi, le coût moyen des pièces observés en séparant la base de donnés avec ce regroupement de marques nous donne :



On obtient ainsi des séries temporelles avec moins de variance qu'avec la répartition par marques, ce qui facilitera la prédiction.
Cependant le résultat n'est pas assez satisfaisant, nous allons donc développer une méthode statistique afin d'obtenir un meilleur regroupement de nos variables explicatives.

3.1 Explication de la méthode générale

3.1.1 Rappel sur l'algorithme des K-moyennes

L'algorithme des k-moyennes est utilisé pour partitionner un ensemble d'observations en un nombre de partitions prédéfinies k. Prenons concrètement une matrice réelle $X \in \mathcal{M}_{n,p}(\mathbb{R})$

$$X = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{np} \end{pmatrix}$$

avec n le nombre d'observations et p le nombre de variables
On initialise en premier lieu k points $\mu_1^0, \mu_2^0, \cdots, \mu_k^0 \in \mathbb{R}$
À chaque étape de l'algorithme, on assimile chaque observation, c'est à dire chaque ligne

$x_i = (x_{i1}, \cdots, x_{ip})$ à son centre point le plus proche Ainsi, $\forall p \in [\![1, k]\!]$, en désignant $S_p^{(t)}$ l'ensemble des points assimilés au point μ_l^t on obtient :

$$S_p^{(t)} = \left\{ x_i : \left\| x_i - \mu_p^{(t)} \right\|^2 \leq \left\| x_i - \mu_p^{(t)} \right\|^2 \ \forall i \in [\![1, n]\!] \right\}$$

Après ça, on recalcule, pour tous les groupes de points, le centre de gravité de chaque groupe, et on définit ainsi les centres de gravité $\mu_i^{(t+1)}$ par

$$\mu_i^{(t+1)} = \frac{1}{|S_i^{(t)}|} \sum_{x_j \in S_i^{(t)}} x_j$$

avec $|S_i^{(t)}|$ le nombre d'éléments de $S_i^{(t)}$.
On répète cette opération jusqu'à que toutes les observations restent aux mêmes centre-points.

Une des principales difficultés est que cet algorithme est très dépendant des points initiaux choisis.
Cependant nous n'avons pas d'autres alternatives que de choisir ces points aléatoirement. Pour éviter le problème de dépendances aux conditions initiales, nous exécutons l'algorithme un certain nombre de fois, en changeant aléatoirement les points initiaux.
Les classes choisit seront celles qui apparaissent le plus de fois.

3.1.2 Méthode utilisée pour notre problème

Nous voulons regrouper, pour chaque variable, les modalités de ces variables en un plus petit nombre de groupes, et ceci en fonction du coût moyen des pièces. Regardons un exemple avec la variable *MARQUE*.

MARQUE	Coût moyen des pièces	Nombre de sinistres
ABA1	543.35 €	9
ABIA	468.27 €	1
AC	949.10 €	1
AIXA	655.03 €	9
ALFA	758.11 €	2 369
ALPI	2 871.57 €	6
AMI1	174.75 €	1
ARO	390 €	1
ASTO	5 584.12 €	44
AUDI	1 092.75 €	16 545
AUST	443.86 €	47
AUVE	620.07 €	2
AVI1	198.72 €	1
BENT	3 619.94 €	52
BMC	438.76 €	1
BMW	1 152.59 €	14 001

Nous avons ici sélectionné uniquement le début du tableau récapitulatif du coût moyen des pièces et du nombre de sinistres par marques.
Nous observons qu'un grand nombre de marques ont une très faible quantité de sinistres,rendant le coût moyen des pièces beaucoup moins fiable.
Une des premières étapes de notre méthode sera donc d'inclure un grand nombre de ces marques dans une marque "Autres", et ceci selon un seuil en terme de nombre de sinistres.
Prenons ce seuil à $n = 10000$.
Nous obtenons ainsi le tableau suivant, trié par nombre de sinistres :

MARQUE	Coût moyen des pièces	Nombre de sinistres
Autres	961.93 €	67 735
RENA	787.88 €	57 926
PEUG	730.90 €	53 020
CITR	730.64 €	37 596
VOLK	830.66 €	24 834
MERC	1 136.65 €	16 952
AUDI	1 092.75 €	16 545
BMW	1 152.59 €	14 001
FORD	733.39 €	13 231
TOYO	802.30 €	11 376
OPEL	740.20 €	10 167

On voit ainsi qu'ici on a appliqué un seuil trop haut, et qu'on a ainsi trop de marques dans la catégorie "Autres". On a donc perdu toute l'information du coût moyen pour les plus petites marques.
La suite de notre étude doit nous permettre de déterminer un seuil optimal.
Une fois notre liste de marques "Autres" créée, nous construisons deux méthodes différentes pour réduire le nombre de modalités de nos variables.

Méthode avec le coût moyen total

La première méthode consiste à appliquer l'algorithme des k-moyennes pour la matrice

$$X = \begin{pmatrix} x_{1,1} \\ x_{2,1} \\ \vdots \\ x_{n,1} \end{pmatrix}$$

avec $x_{i,1}$ le coût moyen des pièces pour la marque i. Cette méthode prend en compte les coûts moyens des pièces calculés de 2011 à 2018. Ainsi, on perd de l'information sur l'évolution du coût moyen par année. Pour contrer cet effet, nous développons une deuxième méthode.

Méthode avec le coût moyen par année

Comme le coût moyen des pièces évolue d'année en année, une deuxième alternative est d'appliquer l'algorithme des k-moyennes pour la matrice

$$X = \begin{pmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,p} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n,1} & x_{n,2} & \cdots & x_{n,p} \end{pmatrix}$$

avec $x_{i,j}$ le coût moyen des pièces pour la marque i et pour la j^{eme} année d'observation

Cependant, dans la méthode des k-moyennes, le nombre de groupes k ne se détermine pas automatiquement.
Il s'agira ainsi d'optimiser ce nombre k, ce qui nous fait deux variables à optimiser : le nombre k de groupes et le seuil à partir duquel on considère qu'une modalité de notre variable doit être dans la catégorie "Autres".
Pour cela, il nous faut une fonction à minimiser en fonction de ces deux variables.

Nous allons donc appliquer un des nos modèles de série temporelle, pour chaque nombre de groupes k dans la méthode des k-moyennes, et pour chaque seuil possible. Nous calculerons ensuite la moyenne des résidus mensuels de ce modèle en valeur absolue.
De façon plus mathématiques, le nombre de cluster optimales k et le seuil optimal s sont estimés par :

$$(\hat{k}, \hat{s}) = \arg \min_{x \in \mathbb{N} y \in \mathbb{N}} f(x, y)$$

Avec

—

$$f(x, y) = \frac{1}{N} \sum_{i=1}^{N} \mid y_i - \hat{y}_i(x, y) \mid \tag{3.1}$$

— $\hat{y}_i(x, y)$ l'estimation du coût moyen des pièces par le regroupement de la variable en question avec le nombre de cluster x et avec le seuil y
— y_i le coût moyen des pièces pour le mois i
— N le nombre de mois pour la création de cluster. Nous prendrons, pour cela, notre base de données de 2011 à 2015. En effet cette création de cluster doit se faire dans notre base de données d'entraînement et non sur notre base de données totale.
Ainsi on a $N = 60$

3.2 Mise en œuvre pratique

Nous effectuerons cette méthode pour chacune de nos variables que l'on veut regrouper, et nous présenterons les résultats ci dessous. Nous utiliserons la méthode ETS pour calculer nos résidus, car elle se révèle être le meilleur compromis entre temps de calcul et efficacité.

Dans la suite de cette partie, on appellera *score* d'une variable, la moyenne des résidus mensuels en valeur absolue du modèle ETS, avec la base de données d'entraînement séparée par la variable en question.

Pour information et à titre de comparaison , le score de la variable ne contenant qu'une seule modalité (ce qui revient en fait à ne pas séparer notre base de données) est de **33.29**.

3.2.1 Variable Marque

Première méthode

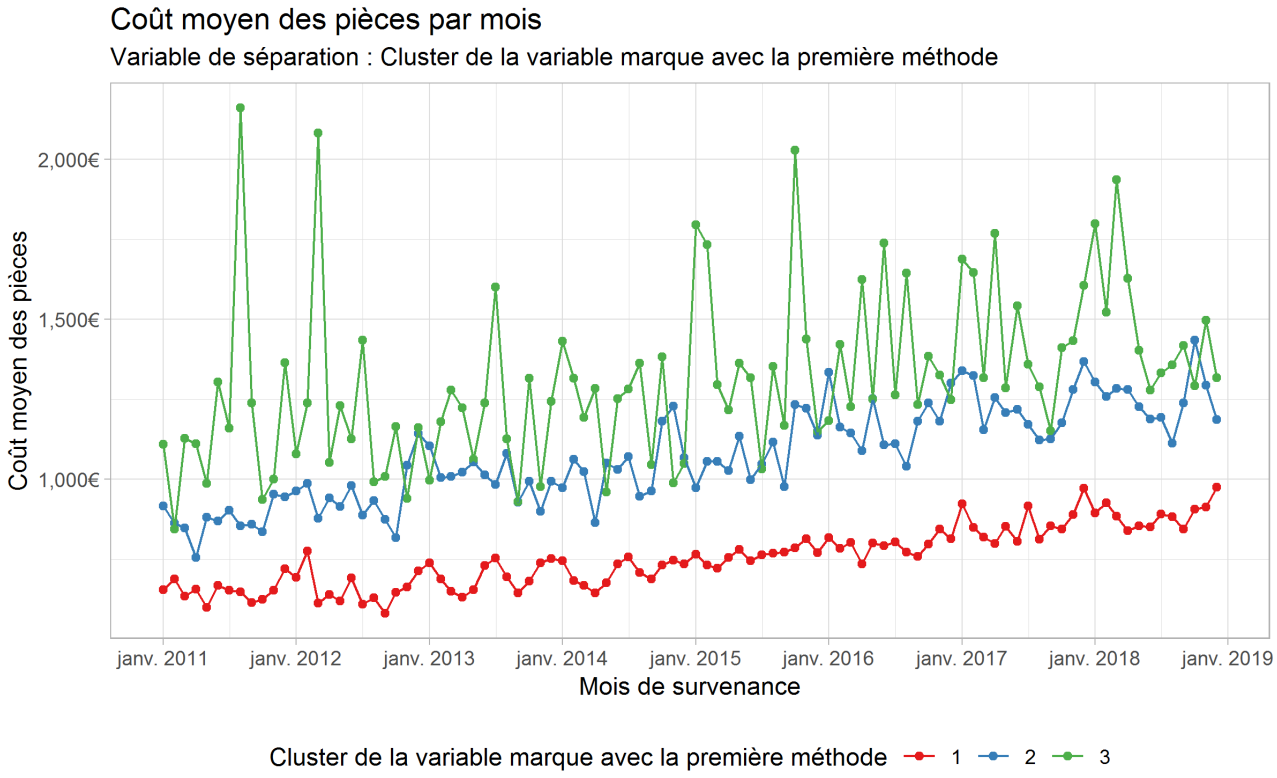
Pour la variable marque, nous obtenons 3 clusters et le seuil des "Autres" a été fixé à 1756 sinistres.

— Dans le premier cluster,sont intégrés les marques :
Alfa Romeo, Citroën, Dacia, Fiat, Ford, Hyundai, Kia, Mini, Opel, Peugeot, Renault, Seat, Skoda, Suzuki, Toyota et Volkswagen.

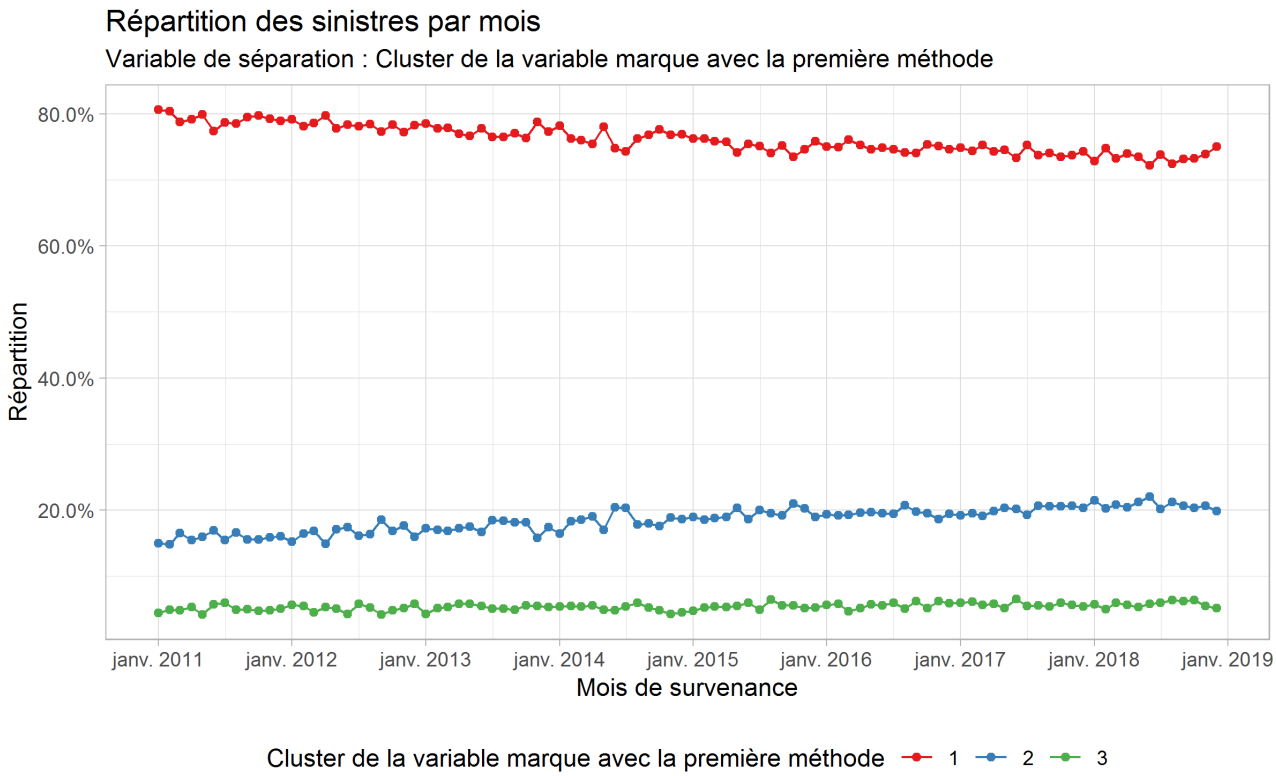
— Dans le deuxième cluster, figurent les marques :
Audi, BMW, Honda, Mercedes-Benz, Nissan, Volvo.

— Dans le dernier cluster, nous retrouverons toutes les autres marques.

Nous obtenons les coûts moyens suivants



et la répartition du nombre de sinistres suivantes :



Nous voyons ainsi que que la répartition des sinistres est stable dans le temps, cependant les trois coûts moyens présentent une forte variance, spécialement pour le cluster *3*

Nous avons en résumé :

Cluster de la variable marque avec la première méthode	Répartition des sinistres	Coût moyen des pièces
1	75.9%	763.04 €
2	18.7%	1 104.21 €
3	5.4%	1 340.24 €

Deuxième méthode

Pour la variable marque, nous obtenons 4 clusters et le seuil des "Autres" a été fixé à 1208

— Dans le premier cluster, nous retrouvons les marques :

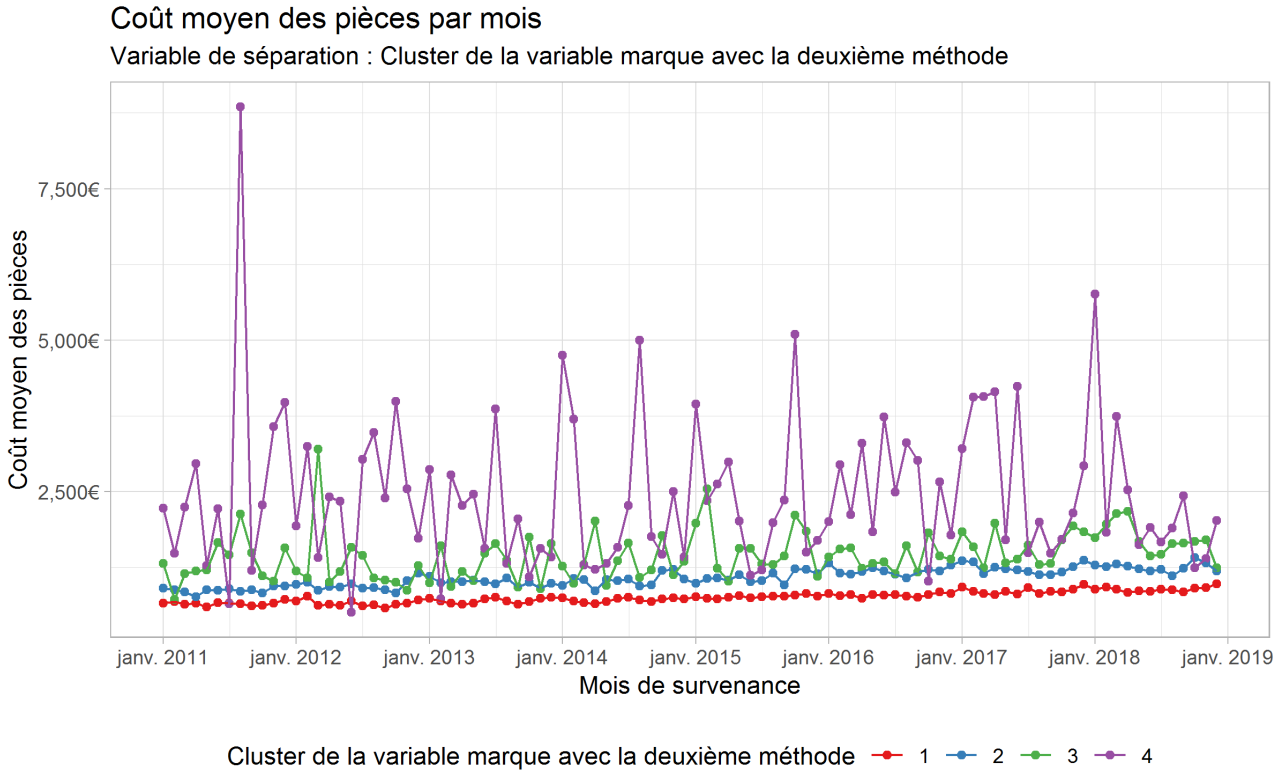
Alfa Romeo,Chevrolet,Citroën, Dacia, Fiat, Ford, Hyundai, Kia, Lancie, Mazda, Mini, Opel, Peugeot, Renault, Seat, Skoda, Smart Suzuki, Toyota et Volkswagen.

— Dans le deuxième cluster, nous retrouvons les marques :
Audi, BMW, Chrysler, Dive, Honda, Mercedes-Benz, Nissan,Mitsubishi Motors,Nissan et Volvo.

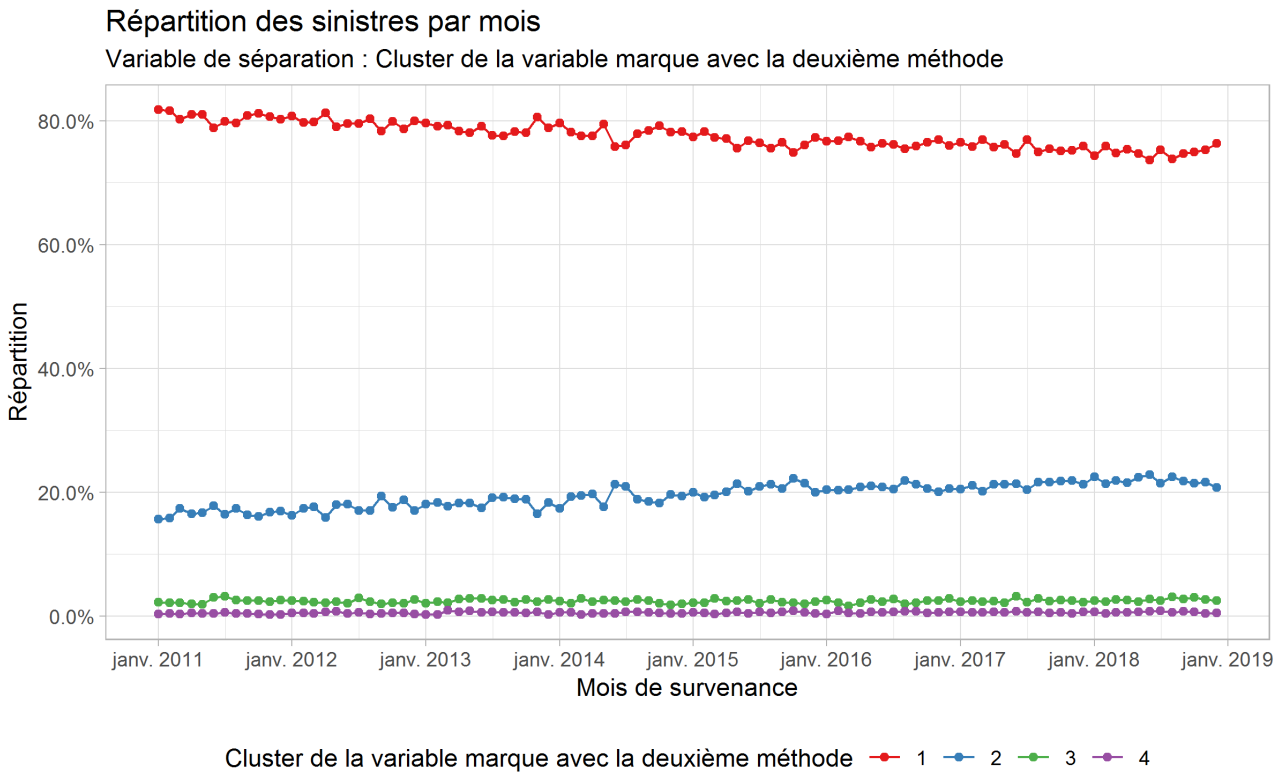
— Dans le troisième cluster, nous retrouvons Jeep, Land Rover et les autres marques.

— Dans le dernier cluster, nous retrouvons la seule marque Porsche

Nous obtenons les coûts moyen suivant :



Nous obtenons aussi la répartition du nombre de sinistres suivante :



et

Cluster de la variable marque avec la deuxième méthode	Répartition des sinistres	Coût moyen des pièces
1	77.3%	763.66 €
2	19.7%	1 107.13 €
3	2.4%	1 471.28 €
4	0.5%	2 521.59 €

Ainsi la deuxième méthode produit un cluster regroupant seulement 0,5% des sinistres. Il y a donc un risque de trouver, pour les mois de la base de données tests, des mois sans sinistres, ce qui pourrait nous causer des erreurs dans nos prédictions.

Conclusion Nous choisirons donc ici la clusturisation proposée par la première méthode. Cette méthode fournit un score de : **33.27**

3.2.2 Variable Kilométrage

Pour la variable kilométrage, nous effectuons tout d'abord un arrondi au millier près de cette variable. De plus nous effectuons le seuillage des "Autres" sur le kilométrage en lui même, et non le nombre de sinistres.

Première méthode Pour la variable kilométrage, nous obtenons 4 clusters et le seuil des "Autres" a été fixé à 120000 km. Ainsi, la variable "Autres" comprends tous les sinistres dont le kilométrage de la voiture est supérieur à 120000 km.

— Dans le cluster 1 , nous retrouvons tous les véhicules avec un kilométrage supérieur à 120000 km

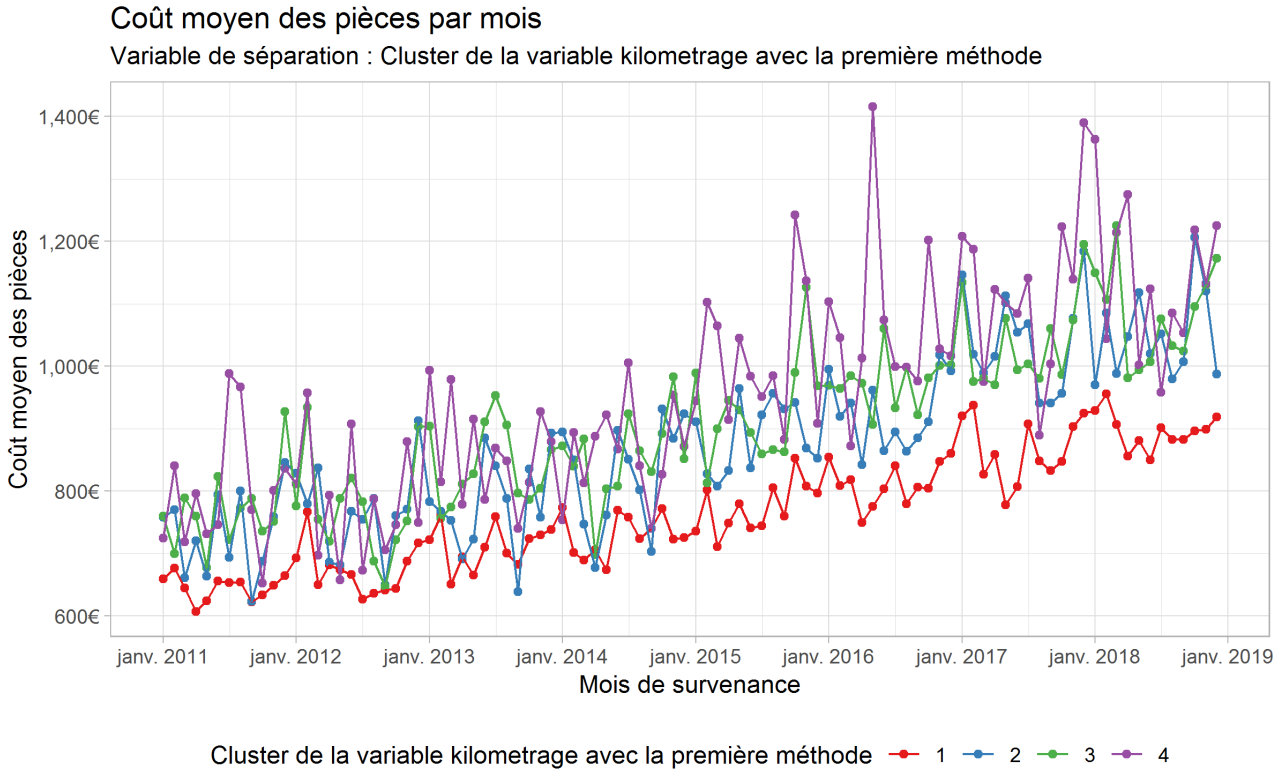
— Dans le cluster 2, apparaissent les véhicules avec un kilométrage compris entre 70000 et 110000 km ainsi que les véhicules avec un kilométrage arrondi à 0 (les véhicules à moins de 5000 km.

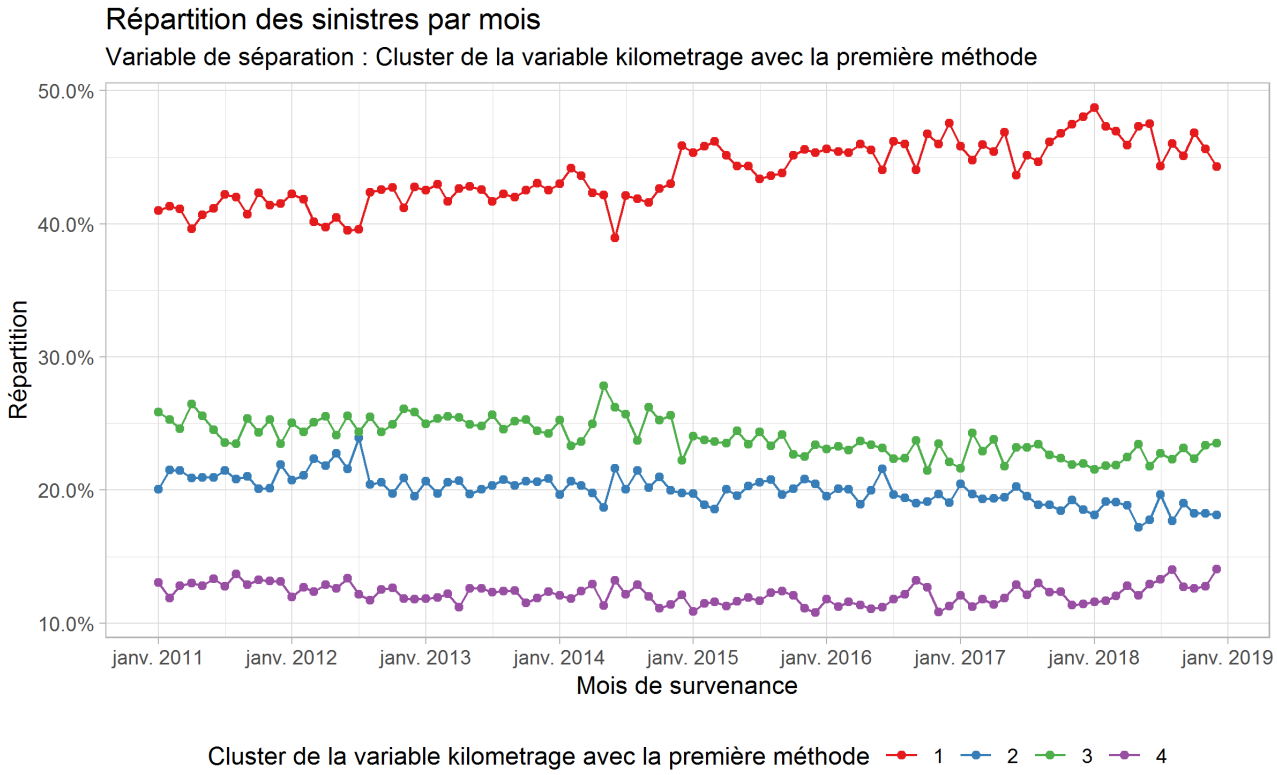
— Dans le cluster 3, sont intégrés les véhicules avec un kilométrage arrondi valant 20000 ou 40000 kilomètres.

— Dans le cluster 4 sont inclus les véhicules avec un kilométrage à 10000 ou 30000 kilomètres.

Le second cluster est satisfaisant, car il contient des véhicules avec des kilométrages cohérents. Le problème vient ici surtout des cluster 3 et 4, pour lesquels on retrouve des véhicules avec un kilométrage de 20000 et 40000 kilomètres pour le cluster 3, et des véhicules avec des kilométrages de 10000 et 30000 kilomètres dans le cluster 4, ce qui manque de cohérence et de continuité.
Ceci est dû à la forte volatilité de nos données

Regardons maintenant ce que donne les coûts moyen des pièces et la répartition des sinistres avec ces clusters.





La répartition des sinistres est stable dans le temps,cependant cette clusturisation sépare la base de données pour donner des séries temporelles avec une forte variance, notamment pour le cluster 4.

Nous avons en résumé :

Cluster de la variable kilometrage avec la première méthode	Répartition des sinistres	Coût moyen des pièces
1	44.1%	779.22 €
2	20.0%	889.13 €
3	23.8%	918.68 €
4	12.2%	974.05 €

Deuxième méthode

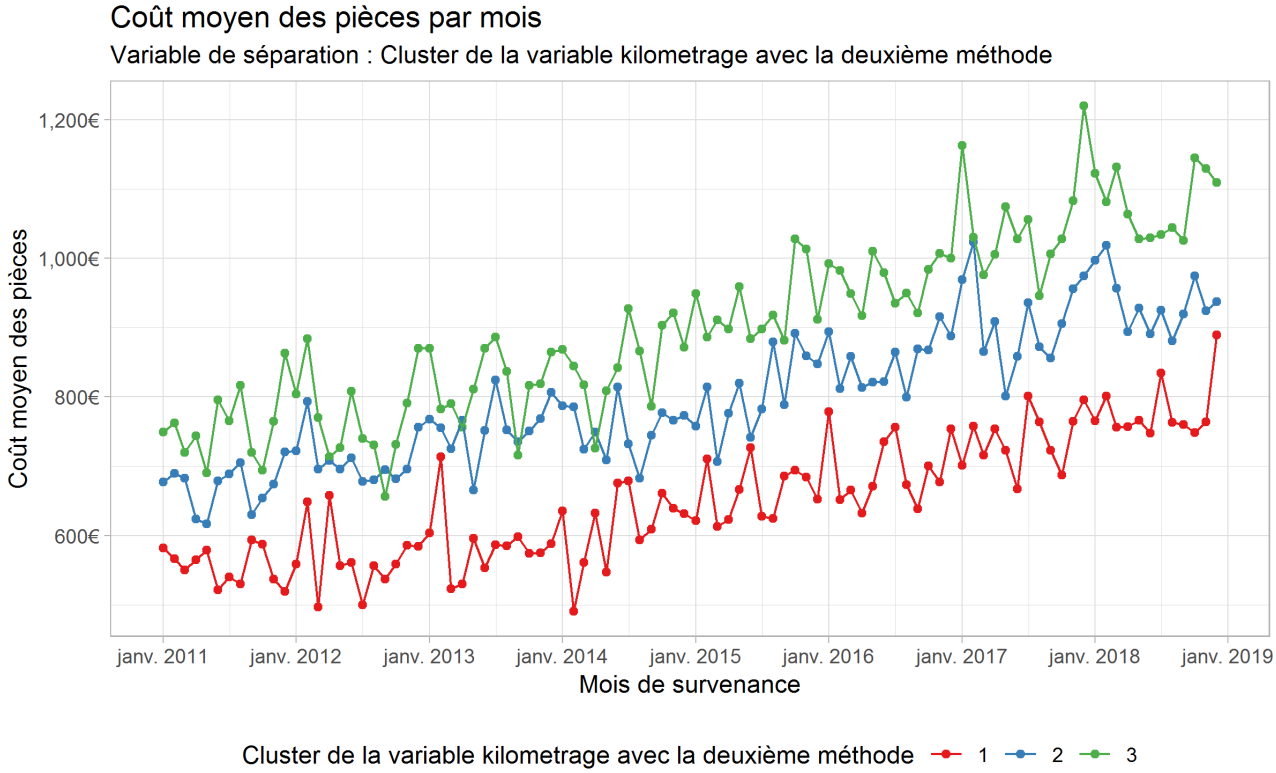
Pour la variable marque, nous obtenons 3 clusters et le seuil des "Autres" a été fixé à 250000 kilomètres

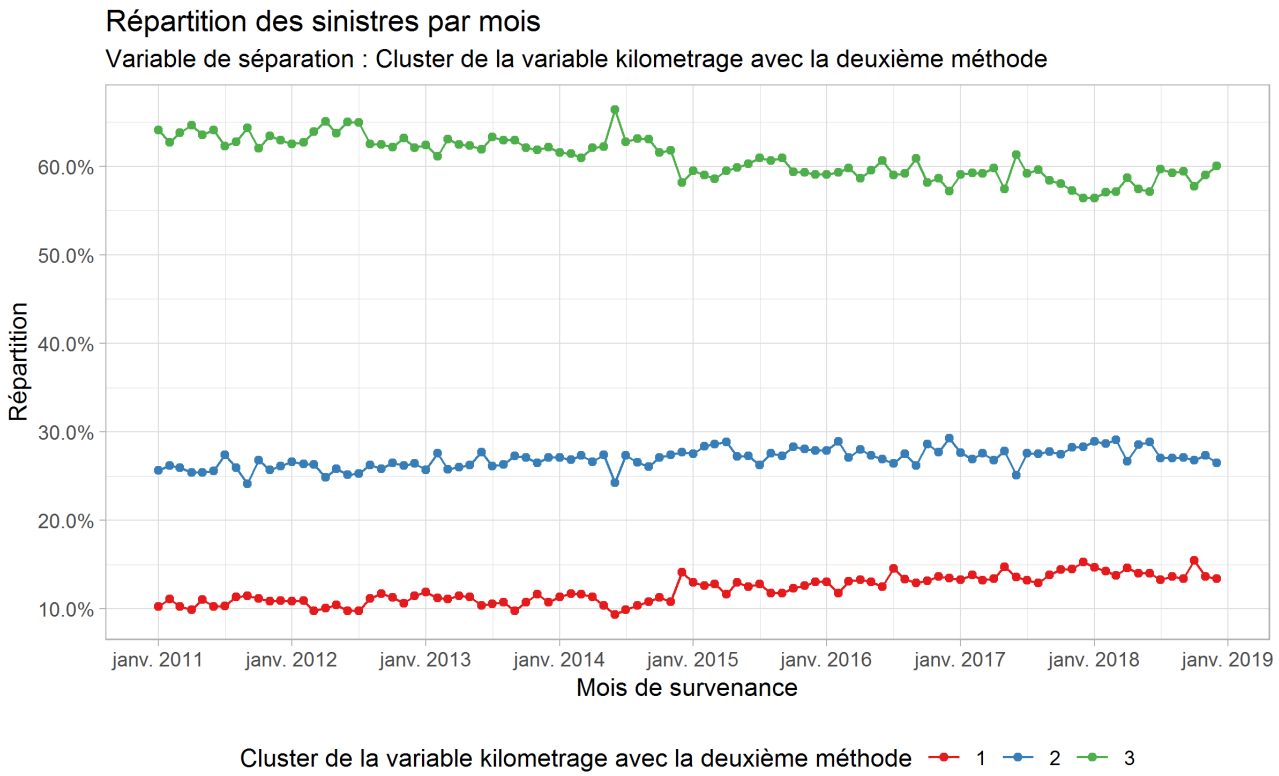
— Dans le premier cluster, nous retrouvons les véhicules avec un kilométrage supérieur à 190000 kilomètres.

— Dans le deuxième cluster, nous retrouvons les véhicules avec un kilométrage compris entre 110000 et 180000 kilomètres.

— Dans le troisième cluster, nous retrouvons les véhicules avec un kilométrage inférieur à 100000 kilomètres.

En visualisant chacun de ces clusters, nous pouvons constater que la répartition apparaît plus équilibré que pour la première méthode, concernant cette variable.
Regardons maintenant la répartition du coût moyen et la répartition des sinistres avec ces clusters.





Nous avons en résumé :

Cluster de la variable kilometrage avec la deuxième méthode	Répartition des sinistres	Coût moyen des pièces
1	12.3%	669.18 €
2	27.0%	814.61 €
3	60.6%	915.86 €

Le coût moyen est ainsi mieux séparé avec cette méthode, et la répartition des sinistres reste stable, nous choisirons donc cette séparation pour la variable kilométrage.

Conclusion : Nous choisirons donc la deuxième méthode. Le score de cette nouvelle variable est de **30.81**

3.2.3 Variable Zonier

Pour la zone dommage, la variable comporte 20 modalités ; de 1 à 20 en fonction du risque dommage en fonction de la localisation du lieu des sinistres ; ainsi qu'une valeur "NAN" pour les sinistres que nous n'avons pas réussi à joindre avec notre zonier. Les 20 classes ont assez d'élément pour que le coût moyen soit significatif, sauf pour la modalité "20" que nous regroupons donc avec les "NAN".
Nous n'effectuons pas de seuillage comme nous le faisions pour les autres variables, dû au faible nombre de modalité dans notre variable de base.

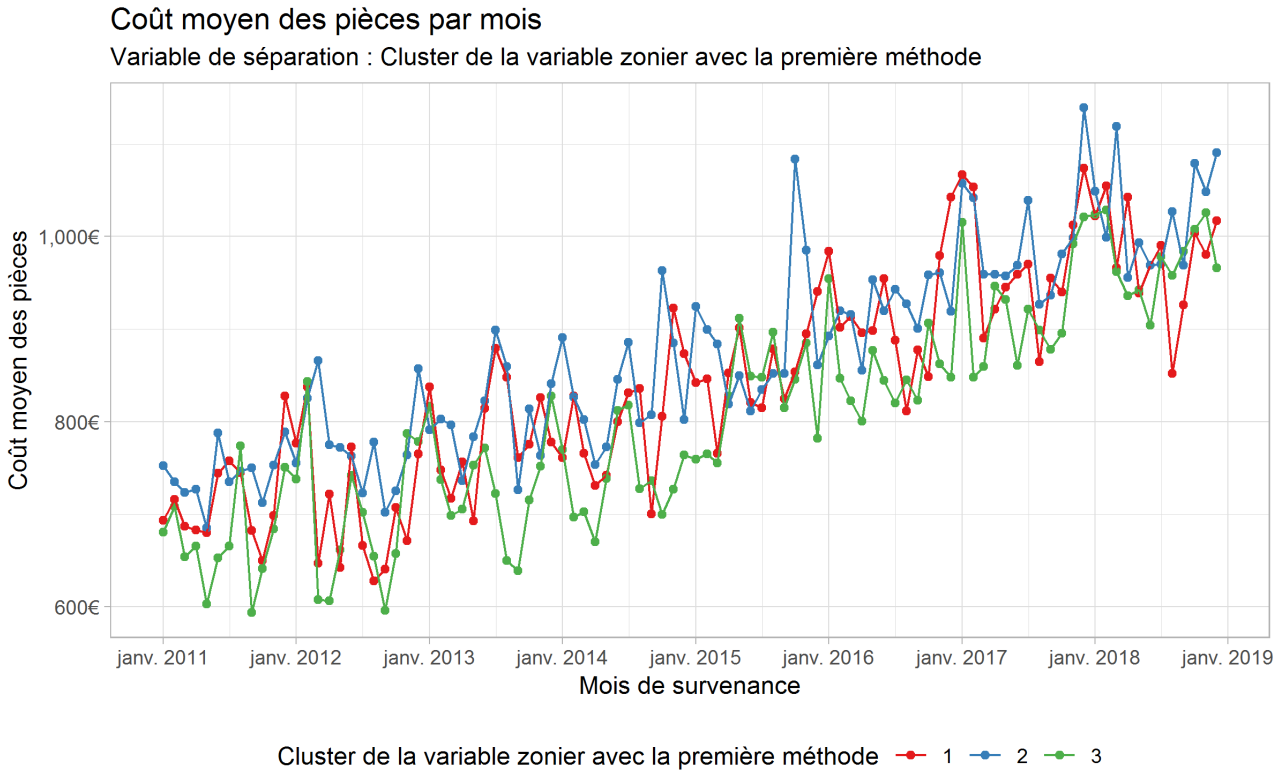
Première méthode

Nous obtenons pour cette méthode, 2 clusters

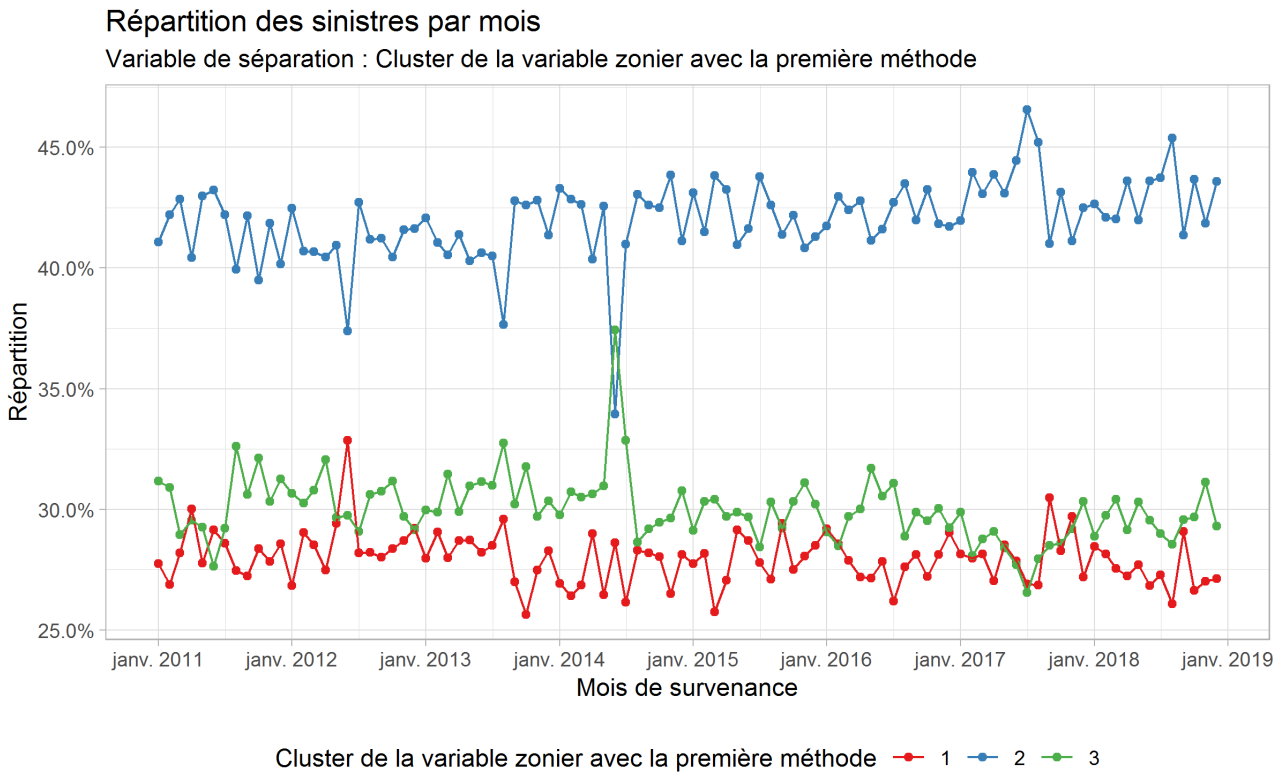
— Dans le premier cluster, nous retrouvons les zones dommages de 1 à 11 ainsi que la zone 13 :

— Dans le second cluster, nous retrouvons les zones, 14 à 19.
— Dans le troisième cluster, nous retrouvons la zone 12 ainsi que la valeur "Nan".

Cette clusturisation apparaît ici assez cohérente, sauf peut être pour la zone 12, mais cela aura un impact négligeable pour la suite.
Nous obtenons les coûts moyens suivant



et la répartition du nombre de sinistres suivantes :



Cette méthode produit 3 séries temporelles présentant toutes les 3 une forte variance. De plus, la répartition des sinistres n'est pas stable dans le temps.

Nous avons en résumé :

Cluster de la variable zonier avec la première méthode	Répartition des sinistres	Coût moyen des pièces
1	28.0%	855.98 €
2	42.1%	888.37 €
3	29.9%	817.46 €

Regardons maintenant les résultats de notre deuxième méthode.

Deuxième méthode

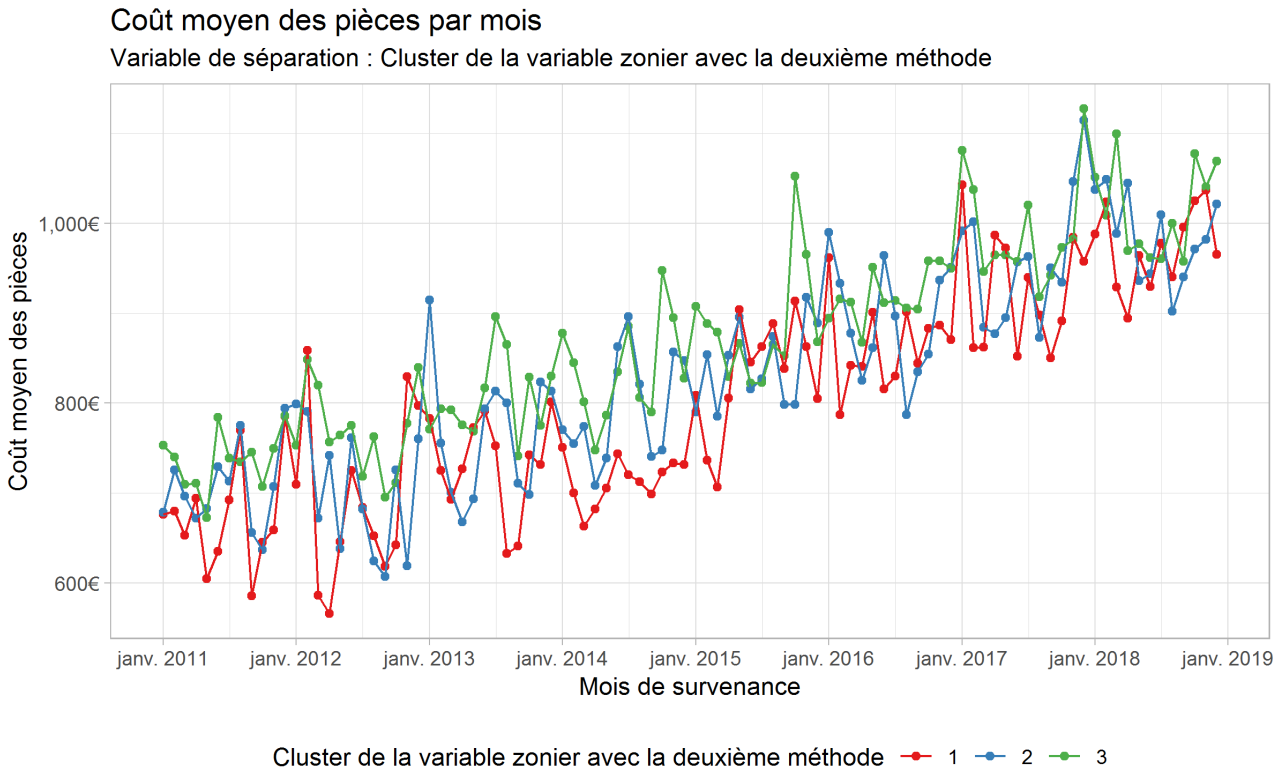
Nous obtenons cette fois ci 3 clusters.

— Dans le premier cluster, nous retrouvons les zones de 1 à 15

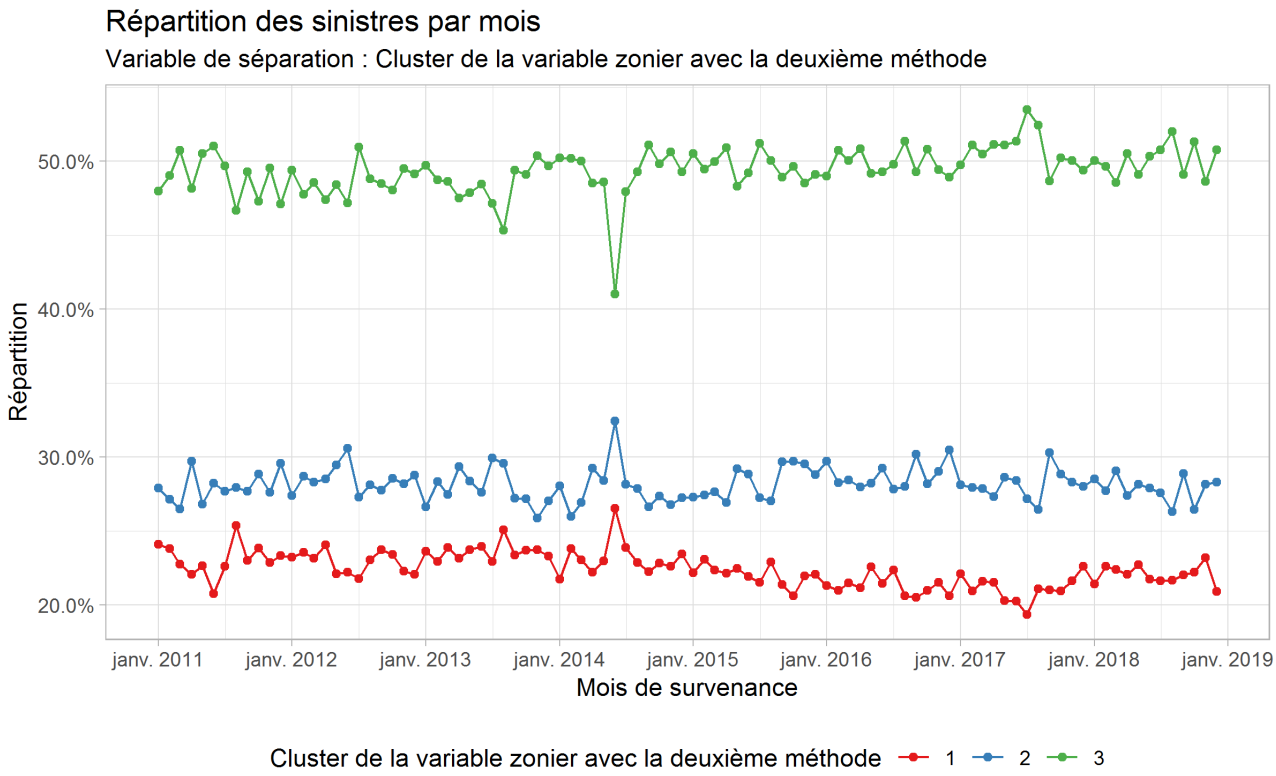
— Dans le deuxième cluster, nous retrouvons la seule zone 19

— Dans le troisième cluster, nous retrouvons les zones de 16 à 18 ainsi que la zone NAN

Nous obtenons les coûts moyen suivants



et la répartition du nombre de sinistres suivantes :



Cette deuxième méthode produit 3 séries temporelles avec une variance plus faible, ainsi qu'une répartition plus stable des sinistres dans le temps.

Nous avons en résumé :

Cluster de la variable zonier avec la deuxième méthode	Répartition des sinistres	Coût moyen des pièces
1	22.3%	814.71 €
2	28.2%	846.33 €
3	49.5%	884.28 €

Conclusion : Nous choisissons encore une fois la deuxième méthode. Le score de cette variable est de **32.21**.

3.2.4 Conclusion

Finalement, nous appliquons ce type de raisonnement pour toutes les variables que l'on clusturise.
On présente les résultats ci dessous, avec le score pour chacune des nouvelles variables.
Nous ajoutons par ailleurs deux variables que l'on ne clusturise pas :
La variable ENERGIE_SRABIS contenant trois modalités :

- GO si la voiture est une voiture gasoil
- ES si la voiture est une voiture essence
- NAN pour les autres voitures

La variable TopAgree contenant deux modalités :

- Agree si le garage réparateur fait partie du réseau de garage agréé de Generali
- NonAgree sinon

Ainsi, chaque variable choisie sera affichée en bleu.

Variable	**Première méthode**		**Deuxième méthode**	
	Score	**Nombre de clusters**	**Score**	**Nombre de cluster**
Âge du vehicule	32.44	6	32.35	5
Classe de réparation	32.84	3	32.84	3
Classe dommage	33.26	5	33.26	5
Groupe SRA	34.37	4	34.37	4
Kilometrage	27.74	4	33.54	3
Marque	33.27	3	31.84	4
Nature sinistre	29.85	3	29.85	3
Nombre de cylindres	32.89	4	32.86	4
Zonier	34.21	3	30.81	3
Puissance administrative	32.57	2	32.57	2
Vitesse maximum	32.95	3	33.72	3
ENERGIE_SRABIS	32.67	3	32.67	3
TopAgree	34.16	2	34.16	2
Total	33.29	1	33.29	1

Chapitre 4

Modèles statistiques

Nous traiterons dans cette partie, les différents modèles théoriques que nous utiliserons. On illustrera cette partie avec notre série temporelle mensuelle des coûts moyens des pièces.

4.1 Généralités sur les séries temporelles

Une série temporelle, ou série chronologique, est un ensemble d'observations pour lesquelles les dates ou elles ont été recueillies jouent un rôle déterminant.

L'analyse de ces séries temporelles peut être poursuivie pour différents buts :
- Description : détermination de composantes...
- Filtrage : transformation de la série dans le but d'éliminer certaines caractéristiques ou des valeurs aberrantes...
- Modélisation : recherche de causalité...
- Prévision

Un des problèmes majeurs est celui de la prévision.
Il s'agit en effet, à partir de valeurs y_1,y_2,...,y_T d'estimer les valeurs y_{T+1},...,y_{T+h}
Évidemment on ne peut pas connaître,en pratique,la dynamique exacte de la série temporelle à travers les valeurs passées. Il y aura toujours une composante aléatoire que l'on ne pourra réduire, et qu'il faudra prendre en compte.
Il faut ainsi voir les nombres y_1,...,y_T comme des réalisations de variables aléatoires $Y_1, ..., Y_T$.
On cherche ainsi à estimer $Y_{t=T+h|t\leq T}$
Afin de prendre en compte cette composante aléatoire, on notera :

$$\forall t \in \mathbf{N} \quad Y_t = d_t + Z_t$$

avec d_t la partie déterministe de notre série temporelle, et Z_t la partie aléatoire appelé bruit.
On peut ensuite décomposer la partie déterministe de façon additive par :

$$\forall t \in \mathbf{N} \quad d_t = m_t + s_t$$

ou m_t représente la tendance, et s_t la saisonnalité

Définition : Saisonnalité
La saisonnalité d'une série temporelle, ou composante saisonnière s_t , correspond à un phénomène qui se répète à intervalles de temps réguliers (périodiques). En général, c'est un phénomène saisonnier d'où le terme de variations saisonnières.
De façon plus mathématiques, on a, pour une saisonnalité de période p :

$$\forall t \in \mathbf{N} \quad s_{t+p} = s_t$$

Il s'agit maintenant d'estimer cette saisonnalité.
Pour cela, il existe plusieurs méthodes, nous présenterons la plus simple ici.

Méthode de la moyenne mobile :
Nous cherchons à estimer une saisonnalité de période 12. Pour cela, nous allons appliquer une moyenne mobile à notre série temporelle.
Notons y_t notre série originale, on distingue alors deux cas :

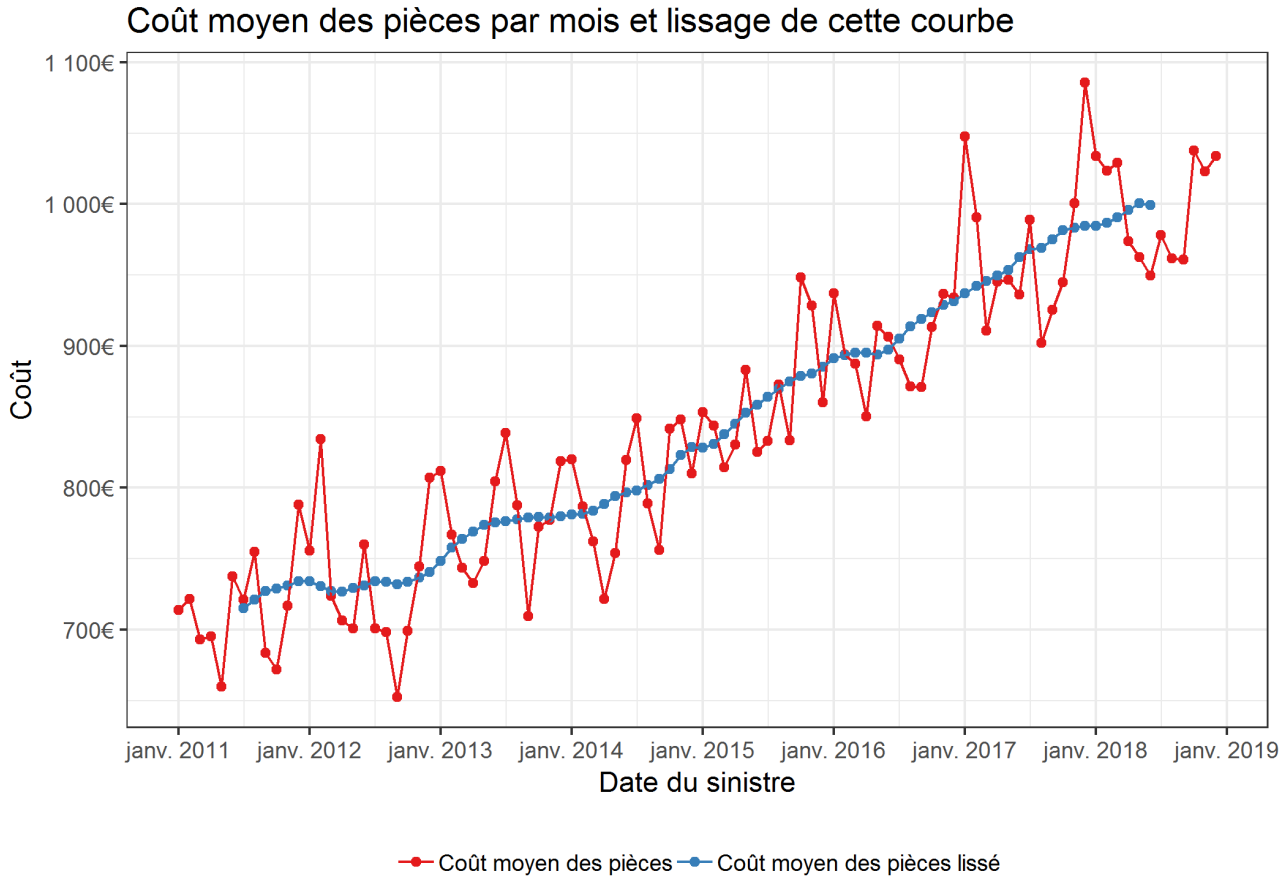
Pour p impair :

$$\forall t \in [\![\frac{p-1}{2}, T-\frac{p-1}{2}]\!] \quad y_t^{MMp} = \frac{1}{p}\sum_{i=t-\frac{p-1}{2}}^{t+\frac{p-1}{2}} y_i \tag{4.1}$$

Pour p pair :

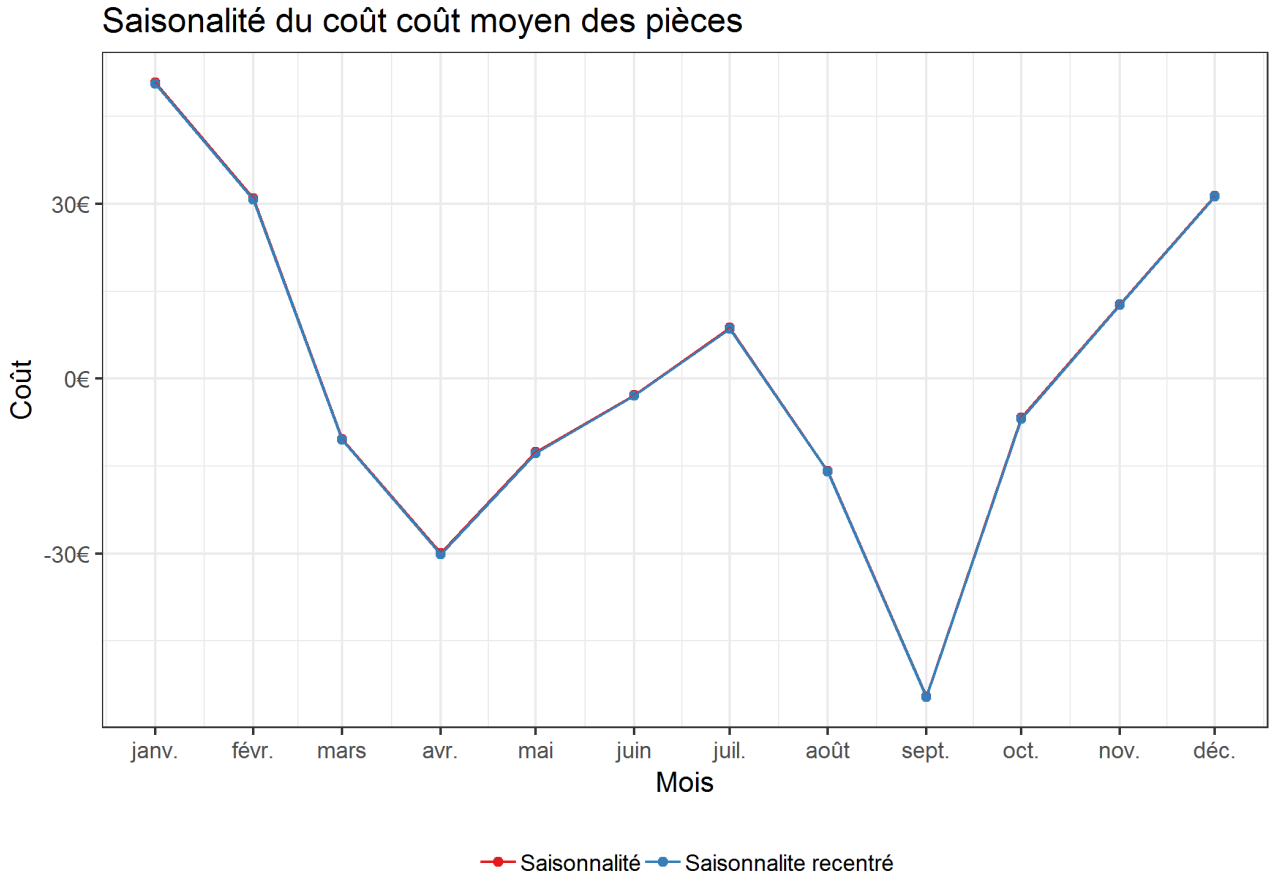
$$\forall t \in [\![\frac{p}{2}-1, T-\frac{p}{2}+1]\!] \quad y_t^{MMp} = \frac{1}{p}(\sum_{i=t-\frac{p}{2}+1}^{t+\frac{p}{2}-1} y_i) + \frac{0.5}{p}(y_{t-\frac{p}{2}} + y_{t+\frac{p}{2}}) \tag{4.2}$$

Concrètement pour notre cas, avec $p = 12$,y_t^{MMp} vaut la moyenne sur les 5 valeurs avant t,t,les 5 valeurs après t ajoutées à la moyenne entre la 6^{me} valeur avant t et la 6^{me} valeur après t.
On réalise ainsi un lissage de notre courbe, ce qui correspondra, à m_t.
Avec notre courbe des coûts moyens des pièces par mois, on obtient :

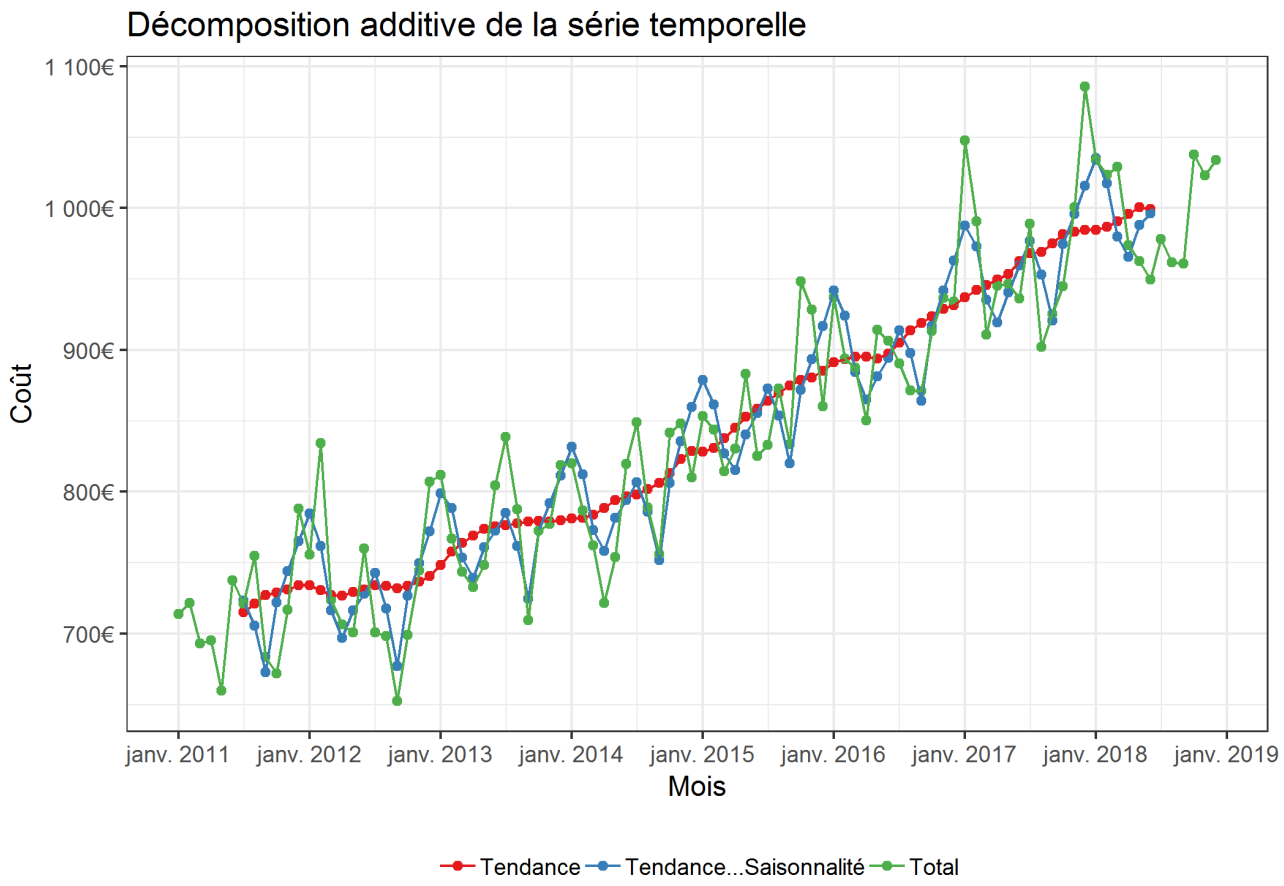


Ensuite , on estime s_t par $y_t - y_t^{MMp}$, puis pour chaque mois de l'année on moyenne y_t pour le mois en question.

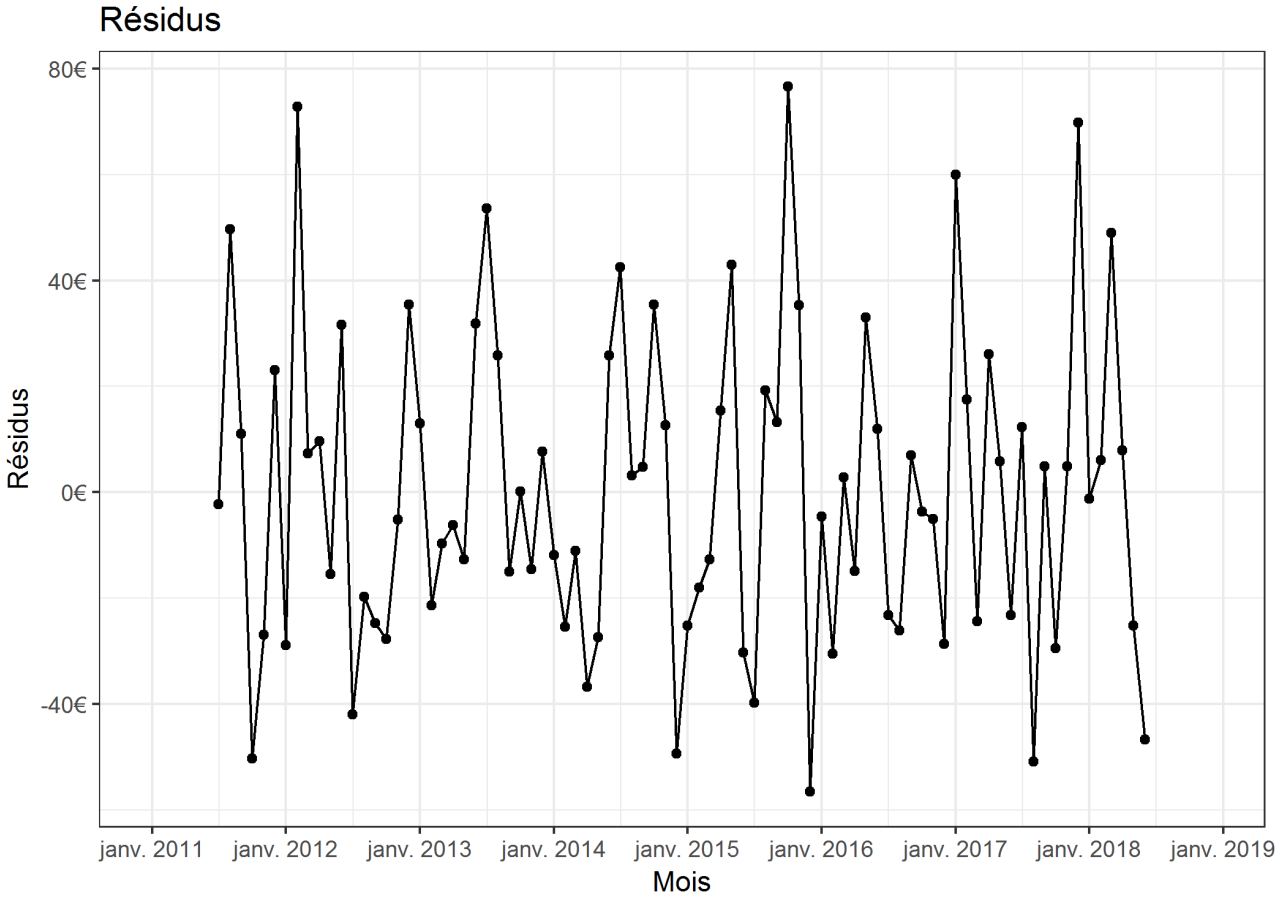
Finalement, on retire la moyenne de ces observations.
On obtient ainsi



On voit ici très peu de différence entre les deux courbes car la moyenne de la saisonnalité avant recentrage était très faible(0.15 €)
On a, au final, la décomposition suivante :



et les résidus suivants :



Cependant, cette méthode présente plusieurs désavantages :

— L'estimation de la tendance (ainsi que des résidus) n'est pas disponible pour les premières et les dernières observations.

— La tendance estimée est souvent trop lissée.

On utilisera ainsi d'autres méthodes de décomposition de la saisonnalité dans nos modèles. Nous détaillerons leurs fonctionnements pour chaque modèles en question.

4.2 Modèle ARIMA

Définition : Série temporelle stationnaires
Une série temporelle stationnaire est une série temporelle dont les propriétés sont indépendantes du temps t où la série est observée. Ainsi, les séries temporelles avec des tendances, ou avec des saisonnalités, ne sont pas stationnaires. En effet,la tendance et la saisonnalité affecteront la valeur de la série temporelle.
En général, une série temporelle stationnaire ne comportera aucun schéma prévisible à long terme. Les graphiques temporels montreront que la série est approximativement horizontale.

4.2.1 Différenciation

En général, les séries temporelles ne sont pas stationnaire. Un moyen efficace de rendre la série stationnaire est de la différentier, c'est à dire regarder la série des différences. Ainsi, pour une différenciation d'ordre 1, on a :

$$\forall t \in [\![2, T]\!] \quad y_t^{(1)} = y_t - y_{t-1} \tag{4.3}$$

Notation :
Pour faciliter la lecture, nous allons introduire l'opérateur *Backshift* B tel que $B(y_t) = y_{t-1}$. Nous avons aussi $B(B(y_t)) = B^2(y_t) = y_{t-2}$
On obtient ainsi :

$$\begin{aligned} \forall t \in [\![2, T]\!] \quad y_t^{(1)} &= y_t - By_t \\ &= (1 - B)y_t \end{aligned} \tag{4.4}$$

Dans le cas où cette différenciation n'est pas suffisante, on peut effectuer une nouvelle différenciation de la série différenciée, pour effectuer ce que l'on appelle une différenciation d'ordre 2.

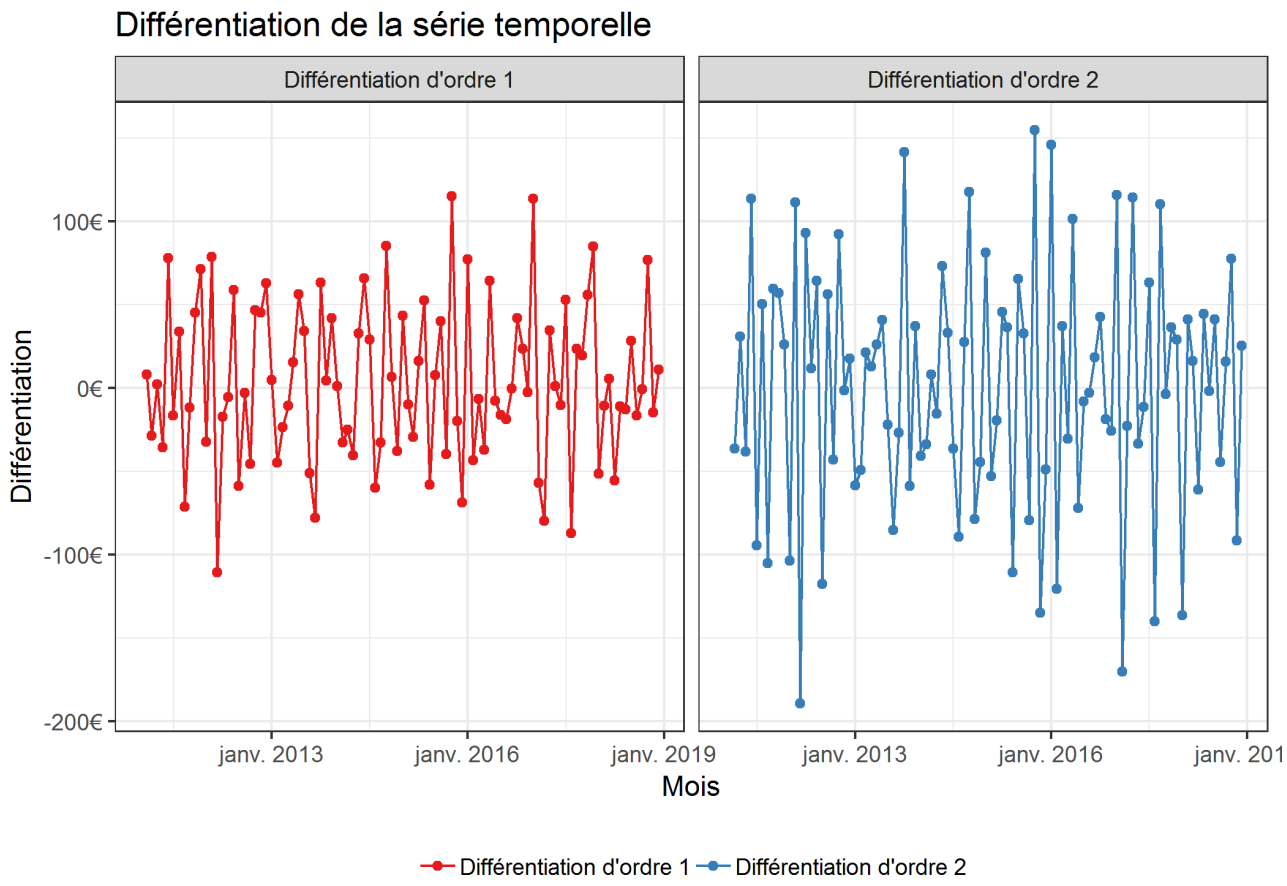
On obtient ainsi :

$$\begin{aligned} \forall t \in [\![3, T]\!] \quad y_t^{(2)} &= y_t^{(1)} - y_{t-1}^{(1)} \\ &= (1 - B)y_t - (1 - B)y_{t-1} \\ &= (1 - B)y_t - (1 - B)By_t \\ &= (1 - B)^2 y_t \\ &= y_t - 2y_{t-1} + y_{t-2} \end{aligned} \tag{4.5}$$

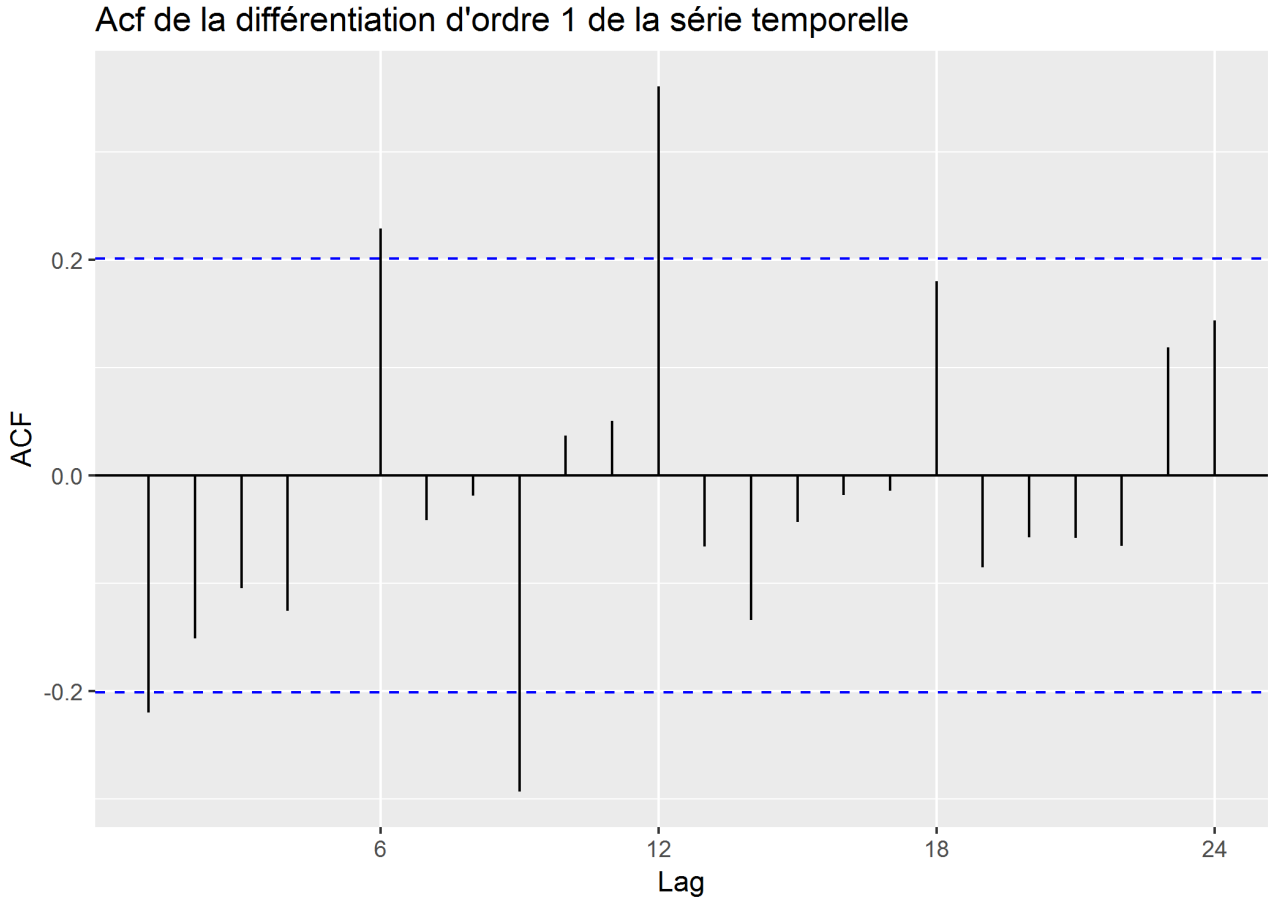
Ainsi une différenciation d'ordre k s'écrira :

$$\forall t \in [\![k + 1, T]\!] \quad y_t^{(k)} = (1 - B)^k y_t \tag{4.6}$$

Pour notre série temporelle, les différenciations d'ordre 1 et 2 donnent :



On voit ainsi ici que la différenciation d'ordre 1 est déjà suffisante, et que au contraire la différenciation d'ordre 2 parait moins adaptée.
Pour vérifier cela, on trace le corrélogramme de notre série différenciée.
On obtient :



On voit ainsi ici que le résultat n'est pas parfait, car la série dépasse à plusieurs reprises les limites bleues, limites selon lesquelles on peut considérer les auto-corrélations comme nulles. Une des raisons principales à ces dépassements est la présence de saisonnalités dans notre série temporelle, saisonnalités qui n'ont pas été effacées par la différenciation.

4.2.2 Modèle ARIMA sans saisonnalité

Un modèle ARIMA est le mélange d'un modèle *AR(Autoregressive models)* avec un modèle *MA(Moving average models)* que nous ne détaillerons pas ici.

Ainsi, un modèle $ARIMA(p,d,q)$ est un modèle pour lequel on modélise y_t par :

$$\begin{aligned} y_t^{(d)} &= c + \phi_1 y_{t-1}^{(d)} + \cdots + \phi_p y_{t-p}^{(d)} + \theta_1 \varepsilon_{t-1} + \cdots + \theta_q \varepsilon_{t-q} + \varepsilon_t \\ (1-B)^d y_t &= c + \sum_{i=1}^{p} (\phi_i (1-B)^d y_{t-i}) + \sum_{j=1}^{q} (\theta_i \varepsilon_{t-i}) + \varepsilon_t \\ (1-B)^d y_t &= c + \sum_{i=1}^{p} (\phi_i (1-B)^d B^i y_t) + \sum_{j=1}^{q} (\theta_i B^i \varepsilon_t) + \varepsilon_t \end{aligned} \tag{4.7}$$

où :

— $c, \phi_1, \cdots, \phi_p, \theta_1, \cdots, \theta_q$ sont des constantes réelles
— $\varepsilon_t, \cdots, \varepsilon_{t-q}$ sont des bruits blancs iid

Ainsi, en factorisant certains termes, on obtient :

$$\begin{array}{ccccc} (1-(\sum_{i=1}^{p}\phi_i B^i)) & & (1-B)^d y_t & = & c+(1+\sum_{i=1}^{q}\theta_i B^i)\varepsilon_t \\ \uparrow & & \uparrow & & \uparrow \\ \text{AR}(p) & & \text{différenciation d'ordre } d & & \text{MA}(q) \end{array} \tag{4.8}$$

4.2.3 Modèle ARIMA avec saisonnalité

Afin de prendre en compte les saisonnalités, on modifie le modèle précédent.
L'idée est de modéliser séparément la dépendance due à la tendance et à la saisonnalité.
Pour cela, il s'agit de remplacer l'opérateur B par B^m avec m le nombre d'observations par an, ici $m = 12$.
On construit ainsi un modèle ARIMA saisonnier $ARIMA(p,d,q)(P,D,Q)_m$ avec :

$$(1-(\sum_{i=1}^{p}\phi_i B^i))(1-(\sum_{i=1}^{P}\Phi_i B^{mi}))(1-B)^d(1-B^m)^D y_t = c+(1+\sum_{i=1}^{q}\theta_i B^i)(1+\sum_{i=1}^{Q}\Theta_i B^{mi})\varepsilon_t \tag{4.9}$$

où :

— $c, \phi_1, \cdots, \phi_p, \theta_1, \cdots, \theta_q, \Phi_1, \cdots, \Phi_P, \Theta_1, \cdots, \Theta_Q$ sont des constantes réelles
— $\varepsilon_t, \cdots, \varepsilon_{t-q}$ sont des bruits blancs iid

4.2.4 Sélection de modèles

Après avoir déterminé la structure de notre modèle, il nous faut maintenant déterminer les différents paramètres de nos modèles.
Pour cela nous allons utiliser la fonction *auto.arima* du package *forecast*, qui lui même choisit les paramètres *d*,*p*,*q*,*D*,*P*,*Q* en minimisant le critère AICc (Akaike's Information Criterion).
Testons ce modèle sur nos données de notre série temporelle pour les années 2011 à 2017 :
Pour le modèle sans saisonnalité, nous obtenons :

— Modèle : ARIMA(0,1,1)
— Coefficients : $\theta_1 = -0.9105, c = 3.5012$
— $AIC = 869.25$ $AICc = 869.55$ $BIC = 876.5$

Nous modelisons ainsi le coût moyen des pièces par mois y_t par :

$$\begin{aligned} (1-B)(y_t - \mu t) &= (1+\theta_1 B)\varepsilon_t \\ y_t - y_{t-1} &= \mu + \varepsilon_t + \theta_1 \varepsilon_{t-1} \end{aligned} \tag{4.10}$$

Nous voyons ainsi que le modèle retient une tendance linéaire pour le coût moyen des pièces.

Pour le modèle avec saisonnalité, on obtient :

— Modèle : $ARIMA(0,1,2)(1,0,0)_m$
— Coefficients : $\theta_1 = -0.5968, \theta_2 = -0.3437, \Phi_1 = 0.5067, \mu = 3.5710$

— $AIC = 844.73 \quad AICc = 845.5 \quad BIC = 856.82$

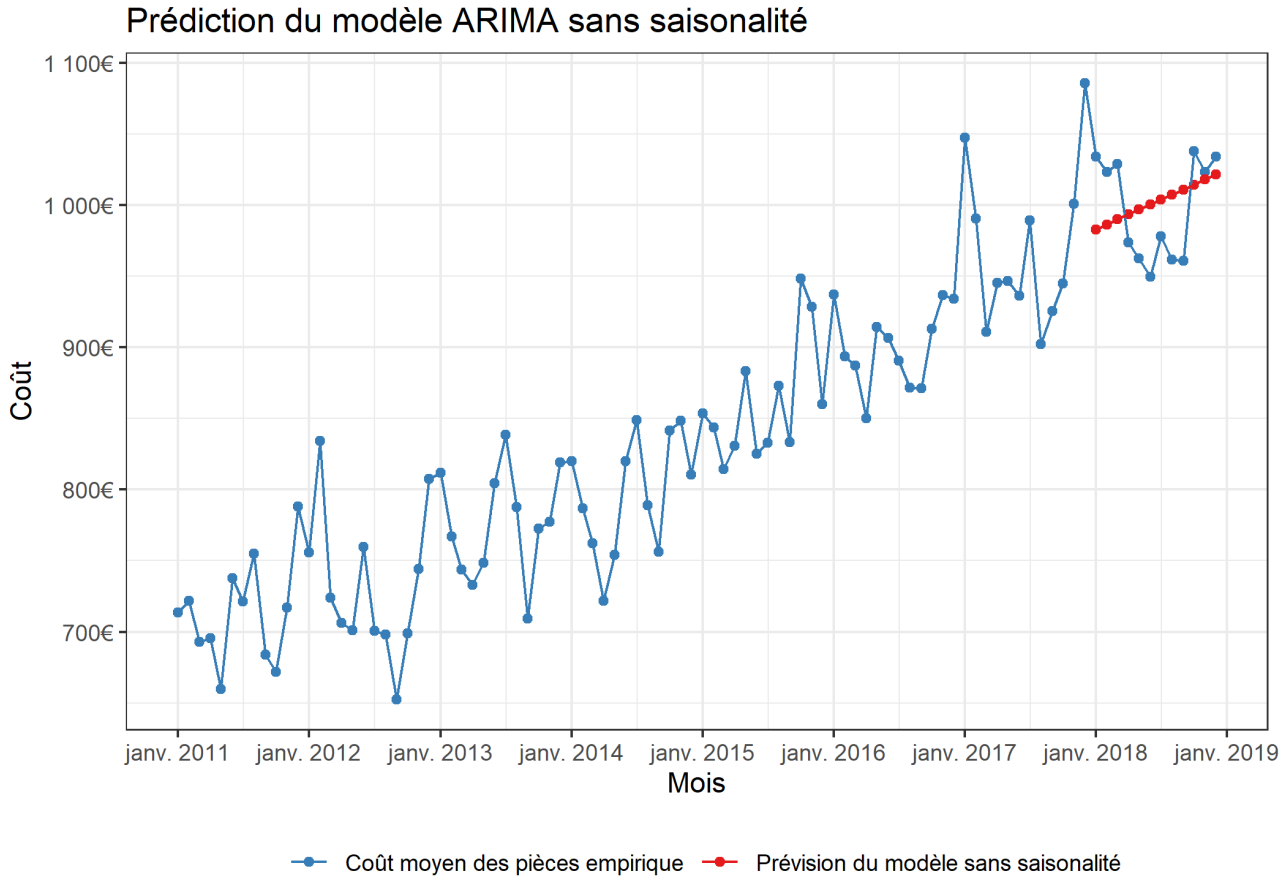
On modélise ainsi le coût moyen des pièces par mois y_t par :

$$\begin{aligned}
&(1-\Phi_1 B^{12})(1-B)(y_t-\mu t) = (1+\theta_1 B+\theta_2 B^2)\varepsilon_t \\
&(1-\Phi_1 B^{12})(y_t-y_{t-1}-(\mu t-\mu(t-1))) = \varepsilon_t+\theta_1\varepsilon_{t-1}+\theta_2\varepsilon_{t-2} \\
&(1-\Phi_1 B^{12})(y_t-y_{t-1}+\mu) = \varepsilon_t+\theta_1\varepsilon_{t-1}+\theta_2\varepsilon_{t-2} \\
&y_t-y_{t-1}+\mu-\Phi_1(y_{t-12}-y_{t-13}+\mu) = \varepsilon_t+\theta_1\varepsilon_{t-1}+\theta_2\varepsilon_{t-2} \\
&y_t-y_{t-1}-\Phi_1 y_{t-12}+\Phi_1 y_{t-13} = \mu(\Phi_1-1)+\varepsilon_t+\theta_1\varepsilon_{t-1}+\theta_2\varepsilon_{t-2}
\end{aligned} \tag{4.11}$$

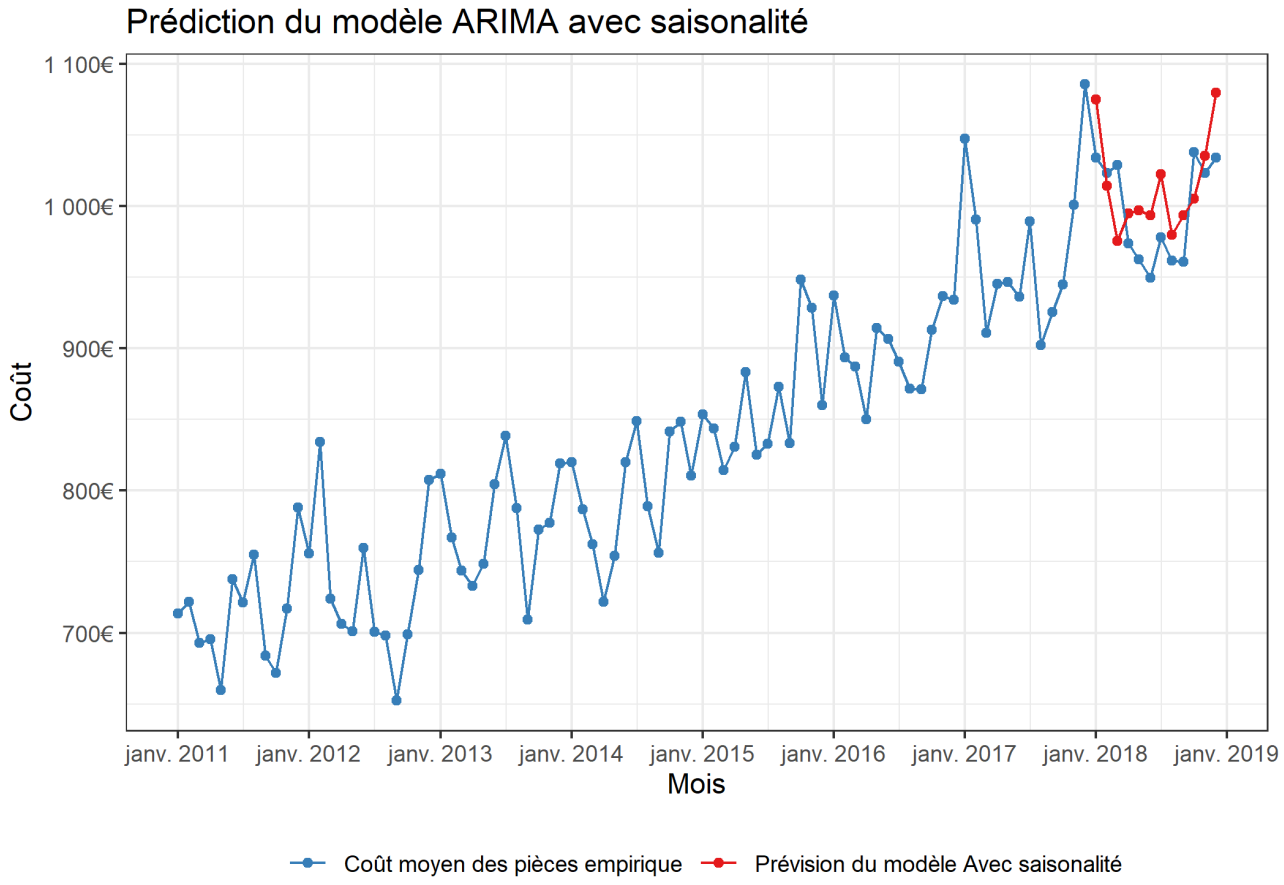
Ici la tendance est plus dure à lire à cause de la saisonnalité de notre série de données.

4.2.5 Résultats

Après avoir estimé les différents paramètres de notre modèle ARIMA, il s'agit maintenant d'utiliser cette paramétrisation pour effectuer nos prédictions.
On présentera les prédictions pour l'année 2018 fourni par la fonction *forecast.arima* du package *forecast*.
Pour la prédiction du modèle sans saisonnalité nous avons :



Nous obtenons sans surprise une tendance linéaire, qui au final cadre assez mal avec nos données de 2018.
Pour la prédiction du modèle avec saisonnalité on a :



Ici, les résultats sont plus satisfaisants et prennent en compte la saisonnalité.

Regardons maintenant l'erreur moyenne absolue, ainsi que l'erreur moyenne absolue en pourcentage des deux modèles :

	Erreur moyenne absolue	Moyenne du pourcentage d'erreur absolue
Modèle sans saisonnalité	32.92€	3.32%
Modèle avec saisonnalité	32.43€	3.25%

‘

L'avantage du modèle avec saisonnalité se confirme ici, même si il est plus faible que prévu et plus faible que ce que l'on peut voir sur le graphique.

4.3 Modèle ETS (Exponential smoothing)

Les modèles ETS (**E**xponen**t**ial **s**moothing) ou ETS(**E**rror,**T**rend,**S**easonal) sont une famille de modèles où la prédiction $\hat{y}_{t+h|t}$ dépend de plusieurs équations d'état, au nombre de 3 maximum,pour modéliser l'erreur (*Error*), la tendance (*Trend*) et la saisonnalité (*Seasonal*).
Nous détaillerons ci-dessous la modélisation de ces 3 composantes.

4.3.1 Modélisation de l'erreur :

Dans un modèle ETS sans partie saisonnalité ni tendance, on a :

$$\text{Équation de prédiction} \qquad \hat{y}_{t+h|t} = \ell_t \tag{4.12}$$

$$\text{Équation de lissage} \qquad \ell_t = \alpha y_t + (1-\alpha)\ell_{t-1} \tag{4.13}$$

avec $\alpha \in [0,1]$.
Cette composition n'est pas particulièrement utile, mais c'est une forme simple qui nous sera utile pour ajouter d'autres composantes de tendance et de saisonnalité par la suite. En particulier, elle nous donne des prédictions constantes suivant h

4.3.2 Modélisation de la tendance :

Méthode d'estimation brute

Pour modéliser la tendance, on prend :

$$\begin{aligned} \text{Équation de prédiction} \qquad & \hat{y}_{t+h|t} = \ell_t + hb_t \\ \text{Équation de lissage} \qquad & \ell_t = \alpha y_t + (1-\alpha)(\ell_{t-1} + b_{t-1}) \\ \text{Équation de tendance} \qquad & b_t = \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)b_{t-1}, \end{aligned}$$

avec $(\alpha, \beta^*) \in [0,1]^2$

Ici, le terme hb_t nous permet de rajouter une tendance linéaire dans nos prédictions, et b_t représente la pente de cette tendance.
Le terme de lissage ℓ_t est une moyenne pondérée de l'observation y_t et de la prédiction de l'équation de prédiction pour $\hat{y}_{t|t-1}$
Le terme de tendance b_t est une moyenne pondérée de la pente estimée au temps t $(\ell_t - \ell_{t-1})$ avec la dernière estimation de la pente b_{t-1}.
Ainsi, la prédiction finale est une fonction linéaire de h.

Méthode d'estimation amortie

La prédiction donnée par la précédente méthode est une courbe avec une pente constante.
Ainsi cette méthode aura tendance à surestimer certaines valeurs, spécialement pour les horizons lointains.
Afin de complexifier le modèle et de remédier à ce problème, on introduit un facteur d'amortissement ϕ tel que :

$$\begin{aligned} \text{Équation de prédiction} \qquad & \hat{y}_{t+h|t} = \ell_t + (\sum_{k=1}^{h} \phi^k) b_t \\ \text{Équation de lissage} \qquad & \ell_t = \alpha y_t + (1-\alpha)(\ell_{t-1} + \phi b_{t-1}) \\ \text{Équation de tendance} \qquad & b_t = \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)\phi b_{t-1}. \end{aligned}$$

avec $\phi \in]0,1[$, $(\alpha, \beta^*) \in [0,1]^2$. À noter que le cas $\phi = 1$ correspond au cas de la méthode d'estimation brute.

Contrairement à la méthode précédente, ici cette méthode produira une estimation constante au bout d'un certain temps.

De façon plus précise on a :

$$\begin{aligned}\lim_{h\to+\infty} \hat{y}_{t+h|t} &= \ell_t + \phi \lim_{h\to+\infty} (\sum_{k=0}^{h-1} \phi^k) b_t \\ &= \ell_t + \frac{\phi}{1-\phi} b_t\end{aligned}$$

4.3.3 Modélisation de la saisonnalité :

On présentera ci-dessous deux méthodes pour modéliser la saisonnalité. La première, additive, conviendra aux saisonnalités constantes dans le temps, tandis que la deuxième,multiplicative, est adaptée quand la saisonnalité change proportionnellement au niveau de la série.

Méthode additive

$$\begin{aligned}\text{Équation de prédiction} \quad & \hat{y}_{t+h|t} = \ell_t + (\sum_{k=1}^{h} \phi^k) b_t + s_{t+h-m(k+1)} \\ \text{Équation de lissage} \quad & \ell_t = \alpha(y_t - s_{t-m}) + (1-\alpha)(\ell_{t-1} + \phi b_{t-1}) \\ \text{Équation de tendance} \quad & b_t = \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*) b_{t-1} \\ \text{Équation de saisonnalité} \quad & s_t = \gamma(y_t - \ell_{t-1} - \phi b_{t-1}) + (1-\gamma) s_{t-m},\end{aligned}$$

avec $\phi \in [0,1]$ (le cas $\phi = 0$ correspondant à l'absence de tendance et $\phi = 1$ le cas de la tendance estimée par méthode brut. On a ensuite $(\alpha, \beta^*) \in [0,1]^2$ et $0 \leq \gamma \leq 1-\alpha$, et k la partie entière de $\frac{h-1}{m}$ avec m la période de notre saisonnalité. Cela nous assure que $t+h-m(k+1) \leq t$.
Ainsi l'équation de lissage est modifiée, on dé-saisonnalise ainsi y_t dans cette équation.
L'équation de tendance est la même que précédemment.
L'équation de saisonnalité est une moyenne pondérée entre la saisonnalité estimée au temps t $(y_t - \ell_{t-1} - b_{t-1})$ et la saisonnalité pour la même période l'année précédente.

En notant que, grâce à l'équation de lissage, on a $\ell_{t-1} + \phi b_{t-1} = \frac{\ell_t - \alpha(y_t - s_{t-m})}{(1-\alpha)}$, on peut transformer l'équation de saisonnalité en :

$$\begin{aligned}\text{Équation de saisonnalité} \quad s_t &= \frac{\gamma}{(1-\alpha)}(y_t - \ell_t) + (1 - \frac{\gamma}{(1-\alpha)})_{t-m} \\ &= \gamma^*(y_t - \ell_t) + (1-\gamma^*) s_{t-m}\end{aligned}$$

avec $0 \leq \gamma^* = \frac{\gamma}{(1-\alpha)} \leq 1$

Méthode multiplicative

Pour la méthode multiplicative, on a :

$$
\begin{aligned}
\text{Équation de prédiction} \quad & \hat{y}_{t+h|t} = [\ell_t + (\sum_{k=1}^{h} \phi^k) b_t] s_{t+h-m(k+1)} \\
\text{Équation de lissage} \quad & \ell_t = \alpha \frac{y_t}{s_{t-m}} + (1-\alpha)(\ell_{t-1} + \phi b_{t-1}) \\
\text{Équation de tendance} \quad & b_t = \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)\phi b_{t-1} \\
\text{Équation de saisonnalité} \quad & s_t = \gamma \frac{y_t}{(\ell_{t-1} + \phi b_{t-1})} + (1-\gamma) s_{t-m}
\end{aligned}
$$

avec $\phi \in [0,1]$ (le cas $\phi = 0$ correspondant à l'absence de tendance et $\phi = 1$ le cas de la tendance estimée par méthode brute.
On a ensuite $(\alpha, \beta^*) \in [0,1]^2$ et $0 \leq \gamma \leq 1 - \alpha$, et k la partie entière de $\frac{h-1}{m}$ avec m la période de notre saisonnalité. Cela nous assure que $t + h - m(k+1) \leq t$.

4.3.4 Modèle statistiques :

Ci-dessus nous avons détaillé plusieurs équations qui sont des équations déterministes qui déterminent juste des prédictions.
Il nous faut ainsi introduire des modèles statistiques qui collent à ces précédentes équations afin de pouvoir générer des intervalles de confiance.
Prenons l'exemple le plus simple : La modélisation simple des erreurs. Pour rappel on avait :

$$
\begin{aligned}
\text{Équation de prédiction} \quad & \hat{y}_{t+h|t} = \ell_t & (4.14) \\
\text{Équation de lissage} \quad & \ell_t = \alpha y_t + (1-\alpha)\ell_{t-1} & (4.15)
\end{aligned}
$$

En prenant $h = 1$ on peut écrire :

$$
\begin{aligned}
\hat{y}_{t+1|t} &= \ell_t \\
&= \alpha y_t + (1-\alpha)\ell_{t-1} \\
&= \alpha y_t + \ell_{t-1} - \alpha \ell_{t-1} \\
&= \ell_{t-1} + \alpha(y_t - \hat{y}_{t|t-1}) \\
&= \ell_{t-1} + \alpha e_t
\end{aligned}
$$

avec $e_t = y_t - \ell_{t-1} = y_t - \hat{y}_{t|t-1}$ le résidu aux temps t.
Ainsi on modélise la première prédiction (qui sera par ailleurs égale à toutes les suivantes) par la fonction de lissage au temps $t-1$ ajouté au résidu du modèle pour l'estimation au temps t (pondérée avec un coefficient α).
Ainsi en supposant que ces résidus sont des bruits blancs iid, on a :

$$
\begin{aligned}
y_t &= \ell_{t-1} + \varepsilon_t & (4.16) \\
\ell_t &= \ell_{t-1} + \alpha \varepsilon_t & (4.17)
\end{aligned}
$$

avec $\forall t \in \mathbb{R} \quad \varepsilon_t \sim \mathcal{N}(0, \sigma^2)$ et les ε_t sont iid

On peut modéliser les erreurs de façon multiplicative comme ci-dessous :

$\varepsilon_t = \frac{y_t - \hat{y}_{t|t-1}}{\hat{y}_{t|t-1}} = \frac{y_t - \ell_{t-1}}{\ell_{t-1}}$
D'où :

$$y_t = \ell_{t-1}(1 + \varepsilon_t) \tag{4.18}$$
$$\ell_t = \ell_{t-1}(1 + \alpha\varepsilon_t). \tag{4.19}$$

De manière plus générale,on obtient :

ADDITIVE ERROR MODELS

Trend	**Seasonal**		
	N	**A**	**M**
N	$y_t = \ell_{t-1} + \varepsilon_t$ $\ell_t = \ell_{t-1} + \alpha\varepsilon_t$	$y_t = \ell_{t-1} + s_{t-m} + \varepsilon_t$ $\ell_t = \ell_{t-1} + \alpha\varepsilon_t$ $s_t = s_{t-m} + \gamma\varepsilon_t$	$y_t = \ell_{t-1}s_{t-m} + \varepsilon_t$ $\ell_t = \ell_{t-1} + \alpha\varepsilon_t/s_{t-m}$ $s_t = s_{t-m} + \gamma\varepsilon_t/\ell_{t-1}$
A	$y_t = \ell_{t-1} + b_{t-1} + \varepsilon_t$ $\ell_t = \ell_{t-1} + b_{t-1} + \alpha\varepsilon_t$ $b_t = b_{t-1} + \beta\varepsilon_t$	$y_t = \ell_{t-1} + b_{t-1} + s_{t-m} + \varepsilon_t$ $\ell_t = \ell_{t-1} + b_{t-1} + \alpha\varepsilon_t$ $b_t = b_{t-1} + \beta\varepsilon_t$ $s_t = s_{t-m} + \gamma\varepsilon_t$	$y_t = (\ell_{t-1} + b_{t-1})s_{t-m} + \varepsilon_t$ $\ell_t = \ell_{t-1} + b_{t-1} + \alpha\varepsilon_t/s_{t-m}$ $b_t = b_{t-1} + \beta\varepsilon_t/s_{t-m}$ $s_t = s_{t-m} + \gamma\varepsilon_t/(\ell_{t-1} + b_{t-1})$
$\mathbf{A_d}$	$y_t = \ell_{t-1} + \phi b_{t-1} + \varepsilon_t$ $\ell_t = \ell_{t-1} + \phi b_{t-1} + \alpha\varepsilon_t$ $b_t = \phi b_{t-1} + \beta\varepsilon_t$	$y_t = \ell_{t-1} + \phi b_{t-1} + s_{t-m} + \varepsilon_t$ $\ell_t = \ell_{t-1} + \phi b_{t-1} + \alpha\varepsilon_t$ $b_t = \phi b_{t-1} + \beta\varepsilon_t$ $s_t = s_{t-m} + \gamma\varepsilon_t$	$y_t = (\ell_{t-1} + \phi b_{t-1})s_{t-m} + \varepsilon_t$ $\ell_t = \ell_{t-1} + \phi b_{t-1} + \alpha\varepsilon_t/s_{t-m}$ $b_t = \phi b_{t-1} + \beta\varepsilon_t/s_{t-m}$ $s_t = s_{t-m} + \gamma\varepsilon_t/(\ell_{t-1} + \phi b_{t-1})$

MULTIPLICATIVE ERROR MODELS

Trend	**Seasonal**		
	N	**A**	**M**
N	$y_t = \ell_{t-1}(1 + \varepsilon_t)$ $\ell_t = \ell_{t-1}(1 + \alpha\varepsilon_t)$	$y_t = (\ell_{t-1} + s_{t-m})(1 + \varepsilon_t)$ $\ell_t = \ell_{t-1} + \alpha(\ell_{t-1} + s_{t-m})\varepsilon_t$ $s_t = s_{t-m} + \gamma(\ell_{t-1} + s_{t-m})\varepsilon_t$	$y_t = \ell_{t-1}s_{t-m}(1 + \varepsilon_t)$ $\ell_t = \ell_{t-1}(1 + \alpha\varepsilon_t)$ $s_t = s_{t-m}(1 + \gamma\varepsilon_t)$
A	$y_t = (\ell_{t-1} + b_{t-1})(1 + \varepsilon_t)$ $\ell_t = (\ell_{t-1} + b_{t-1})(1 + \alpha\varepsilon_t)$ $b_t = b_{t-1} + \beta(\ell_{t-1} + b_{t-1})\varepsilon_t$	$y_t = (\ell_{t-1} + b_{t-1} + s_{t-m})(1 + \varepsilon_t)$ $\ell_t = \ell_{t-1} + b_{t-1} + \alpha(\ell_{t-1} + b_{t-1} + s_{t-m})\varepsilon_t$ $b_t = b_{t-1} + \beta(\ell_{t-1} + b_{t-1} + s_{t-m})\varepsilon_t$ $s_t = s_{t-m} + \gamma(\ell_{t-1} + b_{t-1} + s_{t-m})\varepsilon_t$	$y_t = (\ell_{t-1} + b_{t-1})s_{t-m}(1 + \varepsilon_t)$ $\ell_t = (\ell_{t-1} + b_{t-1})(1 + \alpha\varepsilon_t)$ $b_t = b_{t-1} + \beta(\ell_{t-1} + b_{t-1})\varepsilon_t$ $s_t = s_{t-m}(1 + \gamma\varepsilon_t)$
$\mathbf{A_d}$	$y_t = (\ell_{t-1} + \phi b_{t-1})(1 + \varepsilon_t)$ $\ell_t = (\ell_{t-1} + \phi b_{t-1})(1 + \alpha\varepsilon_t)$ $b_t = \phi b_{t-1} + \beta(\ell_{t-1} + \phi b_{t-1})\varepsilon_t$	$y_t = (\ell_{t-1} + \phi b_{t-1} + s_{t-m})(1 + \varepsilon_t)$ $\ell_t = \ell_{t-1} + \phi b_{t-1} + \alpha(\ell_{t-1} + \phi b_{t-1} + s_{t-m})\varepsilon_t$ $b_t = \phi b_{t-1} + \beta(\ell_{t-1} + \phi b_{t-1} + s_{t-m})\varepsilon_t$ $s_t = s_{t-m} + \gamma(\ell_{t-1} + \phi b_{t-1} + s_{t-m})\varepsilon_t$	$y_t = (\ell_{t-1} + \phi b_{t-1})s_{t-m}(1 + \varepsilon_t)$ $\ell_t = (\ell_{t-1} + \phi b_{t-1})(1 + \alpha\varepsilon_t)$ $b_t = \phi b_{t-1} + \beta(\ell_{t-1} + \phi b_{t-1})\varepsilon_t$ $s_t = s_{t-m}(1 + \gamma\varepsilon_t)$

4.3.5 Estimation des modèles et résultats

Afin d'estimer quel modèle correspond le mieux à nos données, nous allons introduire, comme pour les modèles ARIMA,les différentes mesures de qualité de modèle.
Nous avons, pour les modèles ETS :

— $\mathrm{AIC} = -2\log(L) + 2k$ *Akaike's Information Criterion*
— $\mathrm{AIC_c} = \mathrm{AIC} + \frac{k(k+1)}{T-k-1}$ *Akaike's Information Criterion corrected*

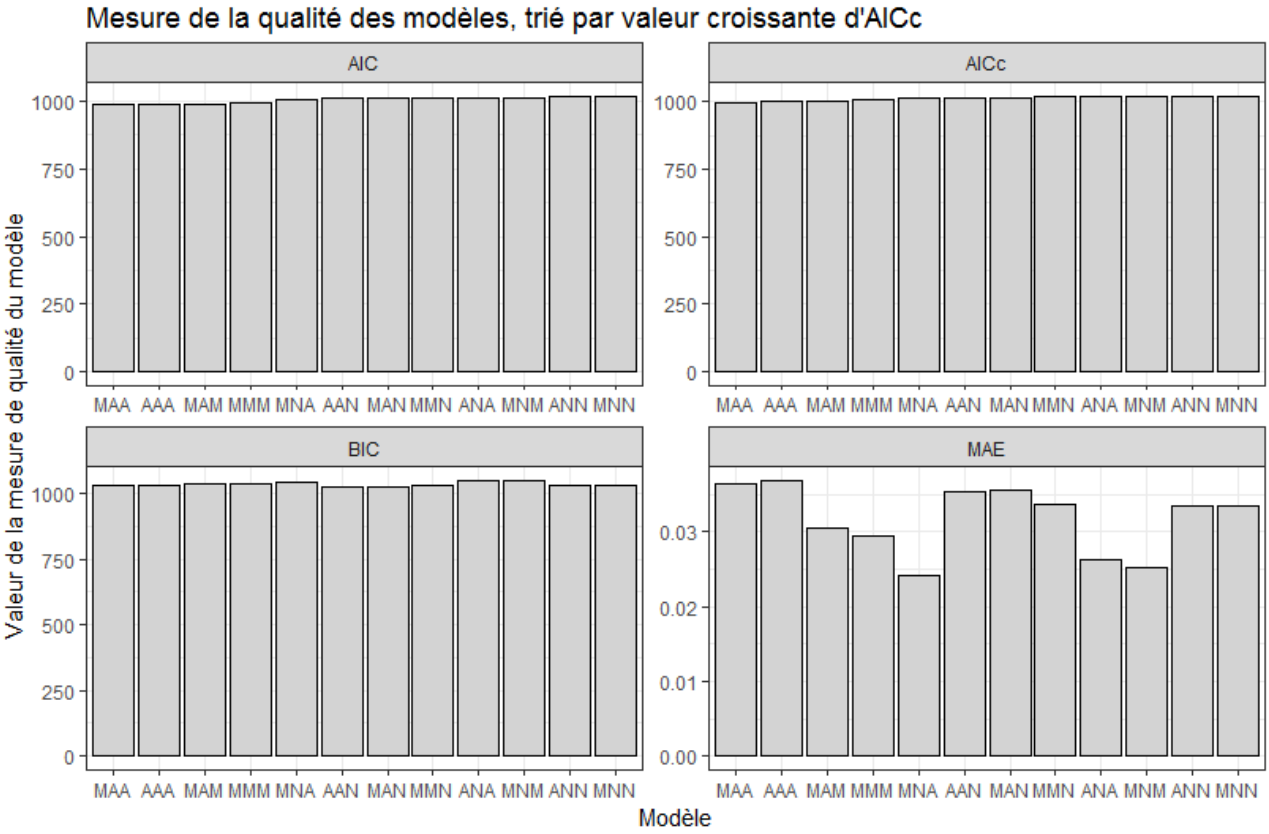
— BIC $= \text{AIC} + k[\log(T) - 2]$ *Bayesian Information Criterion*

Par défaut, la fonction *ets* de R sélectionnera les meilleurs modèles et les meilleurs paramètres du modèle en minimisant le critère AIC_c. Chaque modèle est reperé par trois lettres, représentant respectivement la modélisation choisit pour l'erreur,la tendance, et la saisonnalité. A noter que pour des problèmes de convergence, les modèles suivants sont interdits :

— ANM : Erreur additive,absence de tendance, et saisonnalité multiplicative.
— AAM : Erreur additive,tendance brute,saisonnalité multiplicative.
— AMN : Erreur additive,tendance amortie,absence de tendance.
— AMA : Erreur additive,tendance amortie,saisonnalité additive.
— AMM : Erreur additive,tendance amortie,saisonnalité multiplicative.
— MMA : Erreur multiplicative,tendance amortie,saisonnalité additive.

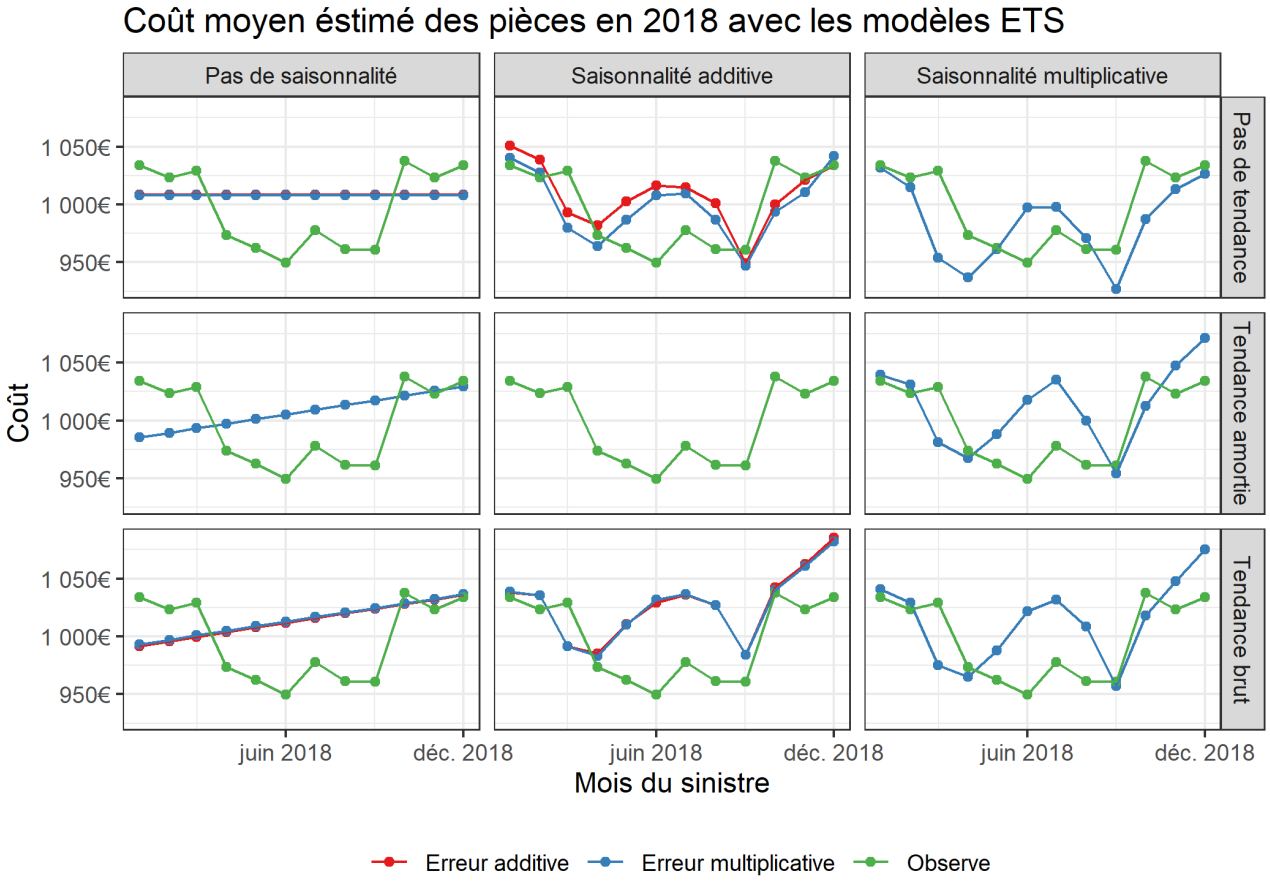
Application à nos données

Ici nous prendrons encore une fois notre série des coûts moyens des pièces par mois de 2011 à 2017 afin de constituer notre série temporelle d'entraînement, pour tester nos modèles sur la série temporelle de 2018.
Nous déterminons ainsi, pour chaque modèle, son AIC,BIC,AICc,calculés sur la base de données d'entraînement, ainsi que le MAE (*Mean absolute error*) , moyenne des erreurs absolues sur la base test.
On trie les modèles par valeur d'AICc croissante, ce qui nous donne :

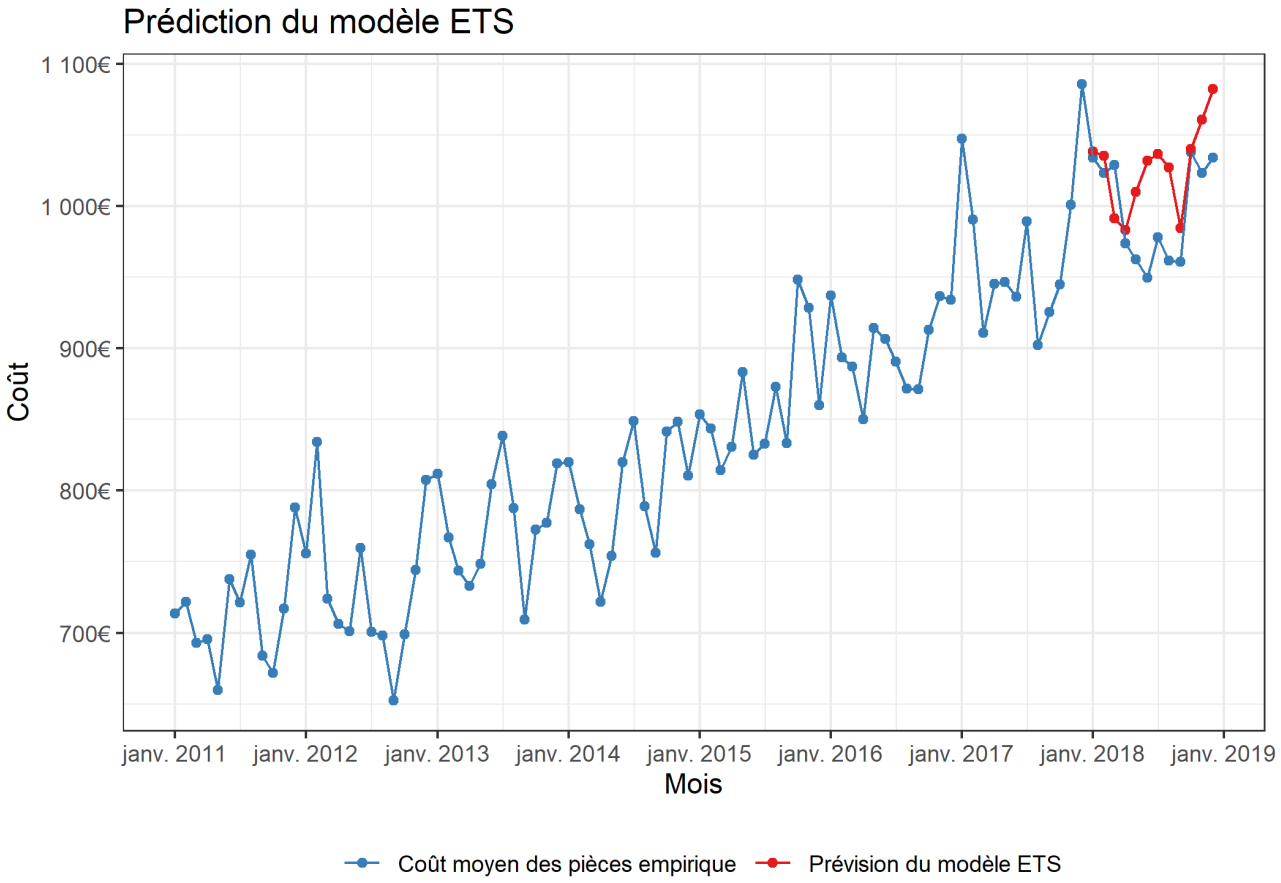


On voit ainsi que le modèle qui nous donne le meilleur BIC (MAA) est un des pires modèles lorsque l'on fait nos prédictions sur la série temporelle test.
Afin de voir tous les comportements de tous les modèles ETS, on trace le coût moyen estimé

par chacun des modèles avec le coût moyen réel des pièces en 2018.
Nous obtenons :



On choisit ainsi le modèle "MAA". Même ci ce modèle sous performe sur la base de données test, cette dernière métrique est moins fiable que l' *AICc* car basé uniquement sur seulement 12 points.
On construit finalement la prédiction finale ainsi que l'intervalle de confiance correspondant avec la fonction *forecast.ets* du package forecast :



Nous avons, en résumé, les résultats suivants :

	Erreur moyenne absolue	Moyenne du pourcentage d'erreur absolue
Modèle ETS	35.77€	3.64%

4.4 TBATS (Trigonometric Seasonal, Box-Cox Transformation, ARMA residuals, Trend and Seasonality)

Le modèle TBATS, dévellopé par De Livera, Hyndman et Snyder en 2011, utilise une transfomation Box-Cox des données, pour ensuite modéliser la série transformée de la même manière que pour les modèles ETS, mais en adoptant une modélisation par série de Fourier pour les éventuelles saisonnalités. Ce modèle permet ainsi de modéliser de façon plus précise les saisonnalités complexes.

4.4.1 Transformation Box-Cox

La transformation de Box-Cox est une transformation non linéaire des données permettant de stabiliser la variance.
Elle est définie comme ceci :

$$\forall t \in \mathbb{R} \quad y_t^{(\omega)} = \begin{cases} \frac{y_t^{\omega}-1}{\omega} & \text{si } \omega \neq 0 \\ \log(y_t) & \text{si } \omega = 0 \end{cases} \tag{4.20}$$

Le réel ω sera choisi, comme les autres paramètres, en minimisant un critère tel que l'AIC, le BIC, ou encore l'AICc.

4.4.2 Algorithme TBATS

On commence par effectuer une transformation Box-Cox des données.
Nous construisons ainsi $y_t^{(\omega)}$ avec :

$$\forall t \in \mathbb{R} \quad y_t^{(\omega)} = \begin{cases} \frac{y_t^{\omega}-1}{\omega} & \text{si } \omega \neq 0 \\ \log(y_t) & \text{si } \omega = 0 \end{cases} \tag{4.21}$$

4.4.3 Modélisation de la saisonnalité

Le modèle TBATS autorise la modélisation de plusieurs saisonnalités, avec des périodes différentes.
Ainsi ce modèle autorise une modélisation avec une saisonnalité de période 12, puis une saisonnalité de période 4.
On note T le nombre de saisonnalités modélisées.
Pour i = 1,2,...T, on modélise la composante saisonnière s_t^i par les trois équations liées suivantes :

$$\begin{cases} s_t^i = \sum_{j=1}^{k_i} s_{j,t}^{(i)} \\ s_{j,t}^{(i)} = s_{j,t-1}^{(i)} + \cos(\frac{2\pi j}{m_i}) + s_{j,t-1}^{*(i)} + \sin(\frac{2\pi j}{m_i}) + \gamma_1^i d_t \\ s_{j,t}^{*(i)} = -s_{j,t-1} \sin(\frac{2\pi j}{m_i}) + s_{j,t-1}^{*(i)} \cos(\frac{2\pi j}{m_i}) + \gamma_2^i d_t \end{cases} \tag{4.22}$$

où

- d_t représente les résidus modélisés par un modèle ARIMA que nous détaillerons après.
- k_i représente le nombre d'harmonique dans la décomposition en série de Fourier de la fonction $s^{(i)} \quad t \mapsto s_t^i$. On prendra, $k_i = \frac{m_i}{2}$ pour m_i pair, et $\frac{m_i-1}{2}$ pour m_i impair
- m_i représente la période de la ième composante saisonnière.
- γ_1^i, γ_2^i représente des paramètres de lissage

4.4.4 Modèle complet

Nous modélisons finalement y_t en effectuant en premier lieu une transformation Box-Cox des données.
Nous construisons ensuite des équations similaires à celles des modèles ETS, en modélisant cependant les résidus d_t avec un modèle $ARIMA(p, q)$, et avec les équations de saisonnalité décrites ci-dessus.
Nous obtenons finalement les équations suivantes :

Transformation Box-Cox $\mathrm{y}_t^{(\omega)} = \begin{cases} \frac{y_t^{\omega}-1}{\omega} & \text{si } \omega \neq 0 \\ \log(y_t) & \text{si } \omega = 0 \end{cases}$

Équation de prédiction $\mathrm{y}_t^{(\omega)} = \ell_{t-1} + \phi b_{t-1} + \sum_{i=1}^{T} s_{t-m_i}^{(i)} + d_t$

Équation de lissage $\ell_t = \ell_{t-1} + \phi b_{t-1} + \alpha d_t$

Équation de tendance $\mathrm{b}_t = (1-\phi)b + \phi b_{t-1} + \beta d_t$

Équation des résidus $\mathrm{d}_t = \sum_{i=1}^{p} \varphi_i d_{t-i} + \sum_{i=1}^{q} \theta_i \epsilon_{t-i} + \epsilon_t$

Équation de saisonnalité $\begin{cases} s_t^i = \sum_{j=1}^{k_i} s_{j,t}^{(i)} \\ s_{j,t}^{(i)} = s_{j,t-1}^{(i)} + \cos(\frac{2\pi j}{m_i}) + s_{j,t-1}^{*(i)} + \sin(\frac{2\pi j}{m_i}) + \gamma_1^{(i)} d_t \\ s_{j,t}^{*(i)} = -s_{j,t-1} \sin(\frac{2\pi j}{m_i}) + s_{j,t-1}^{*(i)} \cos(\frac{2\pi j}{m_i}) + \gamma_2^{(i)} d_t \end{cases}$

Nous avons ainsi à estimer les paramètres suivants :

— $\alpha, \beta, \gamma_1^{(1)}, \gamma_2^{(1)}, ..., \gamma_1^{(T)}, \gamma_2^{(T)}$ des coefficients de lissage
— b le coefficient de tendance long terme
— ϕ le coefficient d'amortissement
— $\varphi_1, ..., \varphi_p, \theta_1, ..., \theta_q$ les coefficients du modèle ARIMA

Après estimation de ces paramètres, nous pouvons recomposer la variable y_t à partir de $y_t^{(\omega)}$ avec une transformée de Box-Cox inversée.
Nous avons aussi $\epsilon_t, \epsilon_{t-1}, ..., \epsilon_{t-q}$ qui sont *iid* et suivent une loi normale centrée de variance σ.
Comme pour les modèles ETS, la fonction *tbats* du packages *forecast* de R détermine automatiquement ces coefficients, en minimisant par défaut l'AICc.
Le modèle TBATS possède plusieurs avantages :

— La transformée Box-Cox permet de modéliser les données non linéaires
— La modélisation ARMA sur les résidus permet de régler les problèmes d'auto-corrélation des résidus
— Permet de modéliser des données avec des saisonnalités multiples

mais aussi plusieurs désavantages :

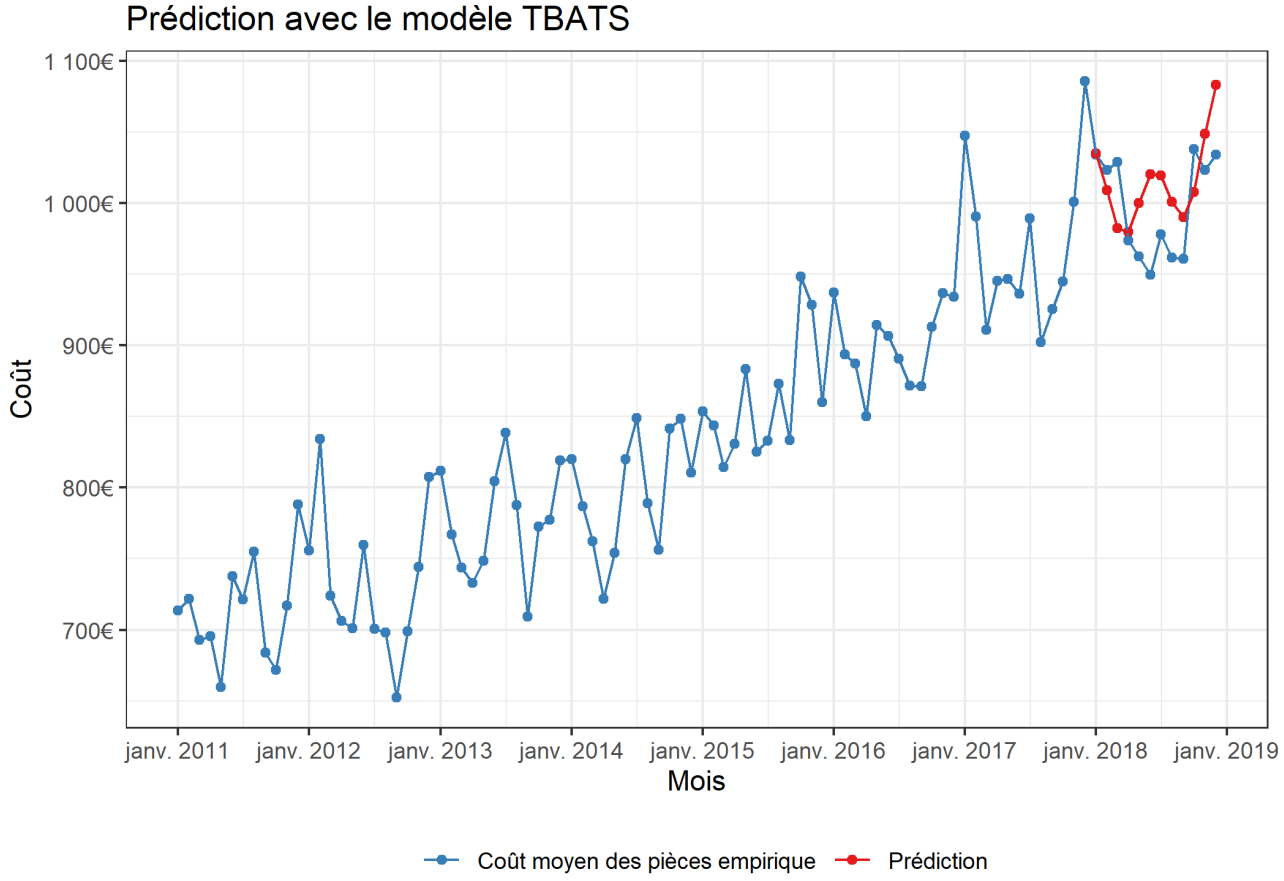
— $\epsilon_t \sim \mathcal{N}(0, \sigma^2)$ n'est pas forcément vrai.
— Les performances pour les prédictions longues se révèlent mauvaises.
— Les temps de calculs sont plus longs que pour les méthodes précédentes.

4.4.5 Résultats

On applique le modèle TBATS sur le coût moyen par mois de 2011 à 2017 pour tester notre modèle sur les données de l'année 2018.
On obtient une modélisation de la série temporelle avec une seule saisonnalité de période 12, avec deux harmoniques. On a aussi $p = q = 0$,ainsi d_t est juste modélisée par une loi normale centrée de variance σ^2.
En résumé on a :

ϕ	α	β	$\gamma_1^{(1)}$	$\gamma_2^{(1)}$	σ
0.574876	0.1483109	-0.0007806816	-0.01577552	0.01166078	1.927295

Pour ce qui est des résultats sur notre base de données test, nous obtenons :



Nous obtenons en résumé :

	Erreur moyenne absolue	Moyenne du pourcentage d'erreur absolue
Modèle TBATS	32,6€	3,29%

4.5 Modèle MLP(Multi Layer Perceptron)

4.5.1 Rappels synthétiques sur les réseaux de neurones

Un réseau de neurones artificiels est un algorithme de machine learning inspiré sur le fonctionnement des neurones biologiques. Ces systèmes autorisent les relations non linéaires entre la variable réponse et les prédictions.
Un réseau de neurones prend en entrée n variables numériques pour en ressortir une ou plusieurs sorties. Un réseau de neurones peut être vu comment un ensemble de neurones simple.

Fonctionnement d'un neurone simple

Un neurone simple prend n valeurs d'entrée que nous allons appeler $x_1, ..., x_n$. Il effectue ainsi une combinaison linéaire de ces entrées, pondérée par des poids $\omega_{j,i}$ appelé poids synaptiques. Afin d'introduire une composante non linéaire, nous appliquons une fonction,appelée fonction d'activation, non linéaire à cette sortie.
Des exemples classiques de fonction d'activation sont :

— La fonction de Heaviside : $H(x) = \begin{cases} 0 & \text{si} \quad x < 0 \\ \frac{1}{2} & \text{si} \quad x = 0 \\ 1 & \text{si} \quad x > 0 \end{cases}$
— La fonction tangente hyperbolique : $\tanh(x) = 1 - \frac{2}{e^{2x}+1}$
— La fonction logistique : $f(x) = \frac{1}{1+e^{-x}}$

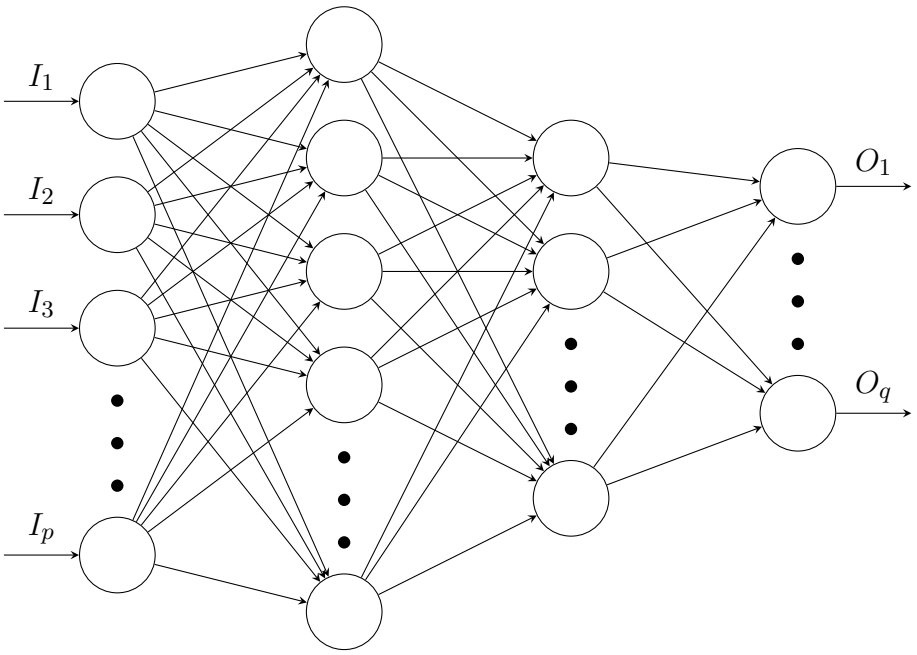
Le neurone j produit ainsi une sortie que l'on notera y_j de la forme :

$$y_j = \varphi(b_j + \sum_{i=1}^{n} \omega_{j,i} x_i)$$

avec φ une fonction d'activation.

Fonctionnement d'un réseau de neurones

Comme son nom l'indique, un réseau de neurones consiste à construire un réseau de neurones simples décrit ci-dessus.
Ainsi le réseau prendra en entrée n valeurs qui constituerons n neurones de départ, situé dans ce que l'on appelle la couche d'entrée. Chacun de ces neurones alimentera l'entrée d'autres neurones, qui sont situé dans ce que l'on appelle la couche cachée. Les sorties de ces neurones alimenteront aussi d'autres neurones de la couche cachée. On poursuit ce processus jusqu'à arriver aux derniers neurones du réseau, les valeurs de sortie de notre système, qui sont dans la couche de sortie.
Il existe ainsi plusieurs architectures pour les réseaux de neurones, c'est à dire plusieurs dispositions de neurones pour la couche cachée,chacune étant utilisée dans des contextes différents. Nous présentons ci dessous l'architecture la plus classique.



Ici, nous avons p neurones d'entrée, q neurones de sortie, ainsi que 2 couches cachées.

Apprentissage du réseau de neurones

Pour p variables d'entrée, et donné $b_1, b_2, ..., b_N, \omega_1, ..., \omega_{x_N}$ avec N le nombre de neurones et x_N le nombre total de poids synaptiques, le réseau sort q variables en sorties.
Maintenant, nous aimerions, suivant une base de données, avec p variables explicatives, estimer au mieux q variables réponses Pour cela, il faut entraîner le réseau, c'est à dire trouver les meilleurs $b_1, b_2, ..., b_N, , \omega_1, ..., \omega_{x_N}$ afin de minimiser l'erreur sur les q variables réponses.
Nous constituons ainsi une base d'entraînement de n lignes, ou, étant donné p variables explicatives, on a les q variables réponses.
L'entraînement d'un réseau de neurones est un processus itératif. On commence par initialiser aléatoirement les paramètres du réseau (c'est à dire les $b_1, b_2, ..., b_N, , \omega_1, ..., \omega_{x_N}$). Ensuite,à chaque observation, c'est à dire à chaque ligne de notre base de données, nous allons ajuster les paramètres du réseau (c'est à dire les $b_1, b_2, ..., b_N, , \omega_1, ..., \omega_{x_N}$) de sorte à réduire l'erreur de prédiction faite par ce même réseau dans l'état actuel. Pour cela, on utilise plusieurs algorithme que nous ne développerons pas ici, comme l'algorithme du gradient ainsi que l'algorithme de rétro-propagation. Pour plus d'informations sur le fonctionnement exact d'un réseau de neurones voir neuralnetworksanddeeplearning.
Nous noterons notamment que l'apprentissage d'un réseau de neurones comporte une partie aléatoire, dans le choix des paramètres de départ. Ainsi, deux entraînements d'un réseau de neurones peut aboutir à deux paramétrages différents.

4.5.2 Fonctionnement théorique des réseaux de neurones pour les séries temporelles

Nous décrirons ci-dessous une application des réseaux de neurones à la prédiction de séries temporelles développée par *Nikolaos Kourentzes* à travers la fonction *MLP* du package R *nnfor*. Cette fonction a été construite grâce aux travaux de *Nikolaos Kourentzesa*, *Devon K. Barrowa* et *Sven F. Cronea* et se base elle même sur la fonction *neuralnet* du package du même nom permettant de construire et d'entraîner des réseaux de neurones.

Nous prendrons encore une fois en exemple la série temporelle des coûts moyens des pièces par mois.
Nous prendrons comme série temporelle d'entraînement cette série temporelle jusqu'à l'année 2017 inclus, et comme série test le coût moyen en 2018.
Nous disposons ainsi de $y_1, y_2, ..., y_t$ et nous voulons prédire $y_{t+1}, y_{t+2}, ..., y_{t+h}$ avec $t = 84$ et

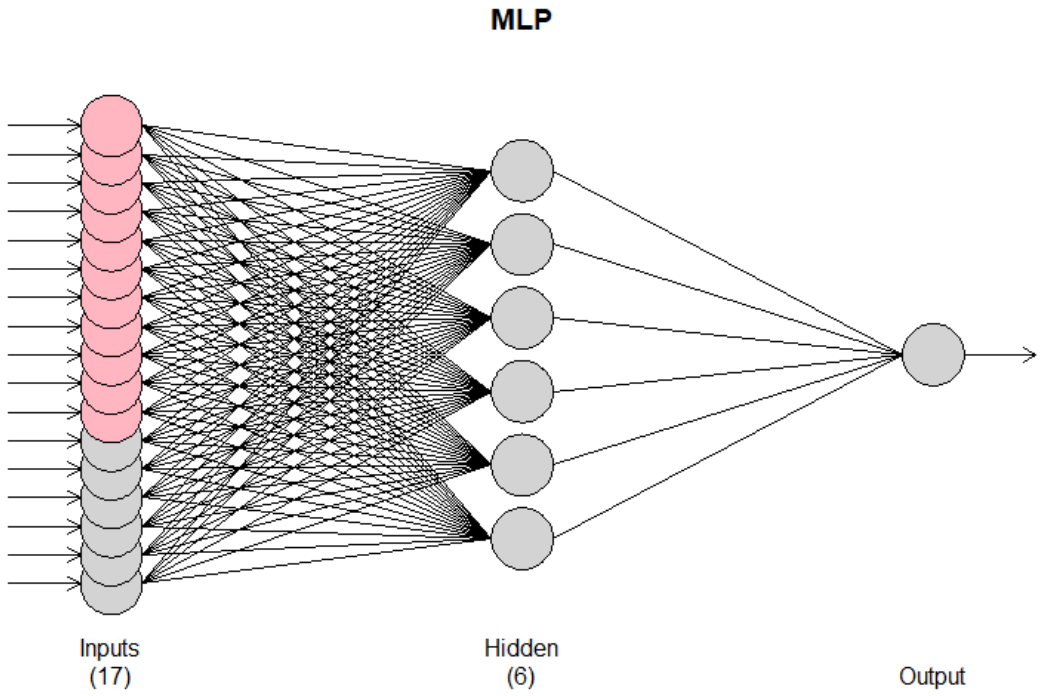
$h = 12$.
L'algorithme commence par retirer la tendance en appliquant une différenciation d'ordre 1, puis on applique une transformation linéaires sur ces données pour ramener la série dans l'intervalle $[-0.8, 0, 8]$.
En effet, des études ont montré que les réseaux de neurones modélisent assez mal les tendances linéaires. On a aussi besoin , pour des questions de stabilité des réseaux de neurones, d'avoir des données d'entrée bornées. Pour plus de lisibilité, nous noterons aussi cette série temporelle transformée y_t.
Pour construire le réseau de neurones nous avons besoin de données d'entrée. Pour cela, nous allons prendre notre série temporelle transformée retardée à l'ordre 1,2,3 etc. Ainsi les premières variables explicatives seront $x_1 = y_{t-1}, x_2 = y_{t-2}, ...x_p = y_{t-p}$. Il s'agit ici de ne pas prendre un trop grand nombre de séries retardées afin d'éviter d'avoir trop de coefficients et d'enlever trop d'observations de notre série temporelle.
Par ailleurs, pour modéliser la saisonnalité, nous introduisons aussi d'autres variables dans notre réseau de neurones,valant respectivement 0 ou 1. Nous verrons en détail ci dessous, dans l'exemple que nous développerons, en quoi l'ajout de ces types de variable permet de modéliser la saisonnalité.
Après l'apprentissage total du réseau de neurones, nous pouvons repasser nos données d'entraînement dans le réseau, et mesurer la moyenne des écarts au carré (MSE). Ainsi, ce MSE nous donne un moyen de juger de la précision d'un réseau de neurones. C'est grâce à cette métrique que nous pouvons déterminer la structure optimale du réseau, c'est à dire la structure des couches cachées. D'après *Nikolaos Kourentzesa*, il est inutile d'avoir plus d'une couche cachée. On se concentrera donc sur des réseaux de neurones et une couche cachée.
Par ailleurs, la fonction d'activation logistique semble être la meilleure, c'est donc cette fonction que nous prendrons.
Nous prendrons comme fonction d'erreur , l'erreur quadratique, c'est à dire
$Erreur(y_t, \hat{y_t}) = \frac{1}{2}(y_t - \hat{y_t})^2$
Il faut bien garder à l'esprit qu'une prédiction à l'aide d'un réseau de neurones reste aléatoire. En effet, les paramètres initiaux sont choisis aléatoirement, et donc deux réseaux de neurones construits sur la même base d'entraînement peuvent être différents.
Pour contrer ce désavantage, nous construisons, pour une série temporelle, un nombre N de réseau de neurones. Les prédictions seront ainsi la médiane des prédictions de ces N réseaux de neurones. Nous étudierons ci dessous dans notre exemple le nombre de réseaux à construire pour avoir une prédiction stable.

4.5.3 Application à nos données

Nous prenons comme série temporelle le coût moyen des pièces de 2011 jusqu'à 2017, pour tester notre modèle sur les données de l'année 2018.
Nous construisons, comme premier exemple un réseau de neurones avec **6** couches cachées.
Nous avons, comme base de données d'entraînement la base suivante :

Y	X1	X2	X3	X4	X5	X9	D1.1	D1.2	D1.3	D1.4	D1.5	D1.6	D1.7	D1.8	D1.9	D1.10	D1.11
0.304343881	-0.100156092	-0.522357521	0.223889849	-0.133471774	0.537185239	0.0403069027	0	0	0	0	0	0	0	0	0	0	1
0.488428515	0.304343881	-0.100156092	-0.522357521	0.223889849	-0.133471774	-0.2192724324	0	0	0	0	0	0	0	0	0	0	0
-0.246863478	0.488428515	0.304343881	-0.100156092	-0.522357521	0.223889849	0.0006388552	1	0	0	0	0	0	0	0	0	0	0
0.542229186	-0.246863478	0.488428515	0.304343881	-0.100156092	-0.522357521	-0.2695516551	0	1	0	0	0	0	0	0	0	0	0
-0.800000000	0.542229186	-0.246863478	0.488428515	0.304343881	-0.100156092	0.5371852394	0	0	1	0	0	0	0	0	0	0	0
-0.139580763	-0.800000000	0.542229186	-0.246863478	0.488428515	0.304343881	-0.1334717743	0	0	0	1	0	0	0	0	0	0	0
-0.055317393	-0.139580763	-0.800000000	0.542229186	-0.246863478	0.488428515	0.2238898488	0	0	0	0	1	0	0	0	0	0	0
0.401370795	-0.055317393	-0.139580763	-0.800000000	0.542229186	-0.246863478	-0.5223575213	0	0	0	0	0	1	0	0	0	0	0
-0.434409735	0.401370795	-0.055317393	-0.139580763	-0.800000000	0.542229186	-0.1001560922	0	0	0	0	0	0	1	0	0	0	0
-0.036037083	-0.434409735	0.401370795	-0.055317393	-0.139580763	-0.800000000	0.3043438812	0	0	0	0	0	0	0	1	0	0	0
-0.340310384	-0.036037083	-0.434409735	0.401370795	-0.055317393	-0.139580763	0.4884285154	0	0	0	0	0	0	0	0	1	0	0
0.315308566	-0.340310384	-0.036037083	-0.434409735	0.401370795	-0.055317393	-0.2468634778	0	0	0	0	0	0	0	0	0	1	0
0.304428415	0.315308566	-0.340310384	-0.036037083	-0.434409735	0.401370795	0.5422291858	0	0	0	0	0	0	0	0	0	0	1
0.430713918	0.304428415	0.315308566	-0.340310384	-0.036037083	-0.434409735	-0.8000000000	0	0	0	0	0	0	0	0	0	0	0
0.016450089	0.430713918	0.304428415	0.315308566	-0.340310384	-0.036037083	-0.1395807628	1	0	0	0	0	0	0	0	0	0	0
-0.333735464	0.016450089	0.430713918	0.304428415	0.315308566	-0.340310384	-0.0553173929	0	1	0	0	0	0	0	0	0	0	0

Nous voyons ainsi qu'on a, en entrée, la série temporelle transformée retardée de 1 jusqu'à **5**,puis la série temporelle retardée de 9, ainsi que 11 variables de saisonnalité, valant 1 avec une fréquence de 12 et 0 sinon.
Ces sélections de retards ont été effectuées par la fonction *mlp* en minimisant le MSE (Mean-squared-error).
On a ainsi cette représentation du réseau de neurones, générée par la fonction *plot.mlp* du package *mlp*, avec, en rouge les neurones d'entrée de saisonnalité, et en gris les retards de la série temporelle transformée.



Pour faciliter la lecture, notons n_r le nombre de séries temporelles retardées en entrée, et n_n le nombre de neurones dans la couche cachée.
Ainsi, on a ici $n_r = 6$ et $n_n = 9$.
On a donc, en sortie du i^{eme} neurones :

$$h_i = f(b_i + \sum_{j=1}^{n_r} \omega_{j,i} y_{t-r_j} + \sum_{k=1}^{11} \omega^*_{k,i} s_k)$$

avec

- f la fonction logistique c'est à dire $f(x) = \frac{1}{1+\exp -x}$.
- r_j les index des retards, soit $r_1 = 1, r_2 = 2, r_3 = 3, r_4 = 4, r_5 = 5, r_6 = 9$.
- s_k la variable D1.k dans la base de données ci dessus. Cette variable vaut 1 périodiquement, avec une période 12.

On a ainsi, dans notre cas précis :

$$\forall i [\![1; 7]\!] \quad h_i = \frac{1}{1 + e^{-(b_i + \sum_{j=1}^{6} \omega_{j,i} y_{t-r_j} + \sum_{k=1}^{11} \omega^*_{k,i} s_k)}}$$

Ainsi, les $\omega^*_{k,i}$ représentent ce qu'on doit ajouter ou soustraire pour le mois de prédiction en question. Il modélise donc la saisonnalité. Par ailleurs, nous avons seulement 11 $\omega^*_{k,i}$ pour des questions d'identifiabilité, comme c'est le cas par exemple lorsque l'on traite des variables qualitatives dans un GLM.
La prédiction finale est obtenue en faisant une combinaison linéaire des n_n neurones cachés, sans application de la fonction d'activation. On a ainsi :

$$\begin{aligned} \hat{y_t} &= b + \sum_{i=1}^{n_n} \Omega_i h_i \\ &= b + \sum_{i=1}^{n_n} \frac{\Omega_i}{1 + e^{-(b_i + \sum_{j=1}^{n_r} \omega_{j,i} y_{t-j} + \sum_{k=1}^{11} \omega^*_{k,i} s_{k,i})}} \end{aligned}$$

Apprentissage du réseau

Dans cette partie, nous simplifierons les notations, notamment pour les variables d'entrées, en notant x la variable d'entrée tel que :

$$x = \begin{pmatrix} x_1 \\ \vdots \\ x_{n_r} \\ x_{n_{r+1}} \\ \vdots \\ x_{n_x} \end{pmatrix} = \begin{pmatrix} y_{t-r_1} \\ \vdots \\ y_{t-r_j} \\ s_1 \\ \vdots \\ s_{11} \end{pmatrix}$$

Avec n_x le nombre d'entrées et $n_x = n_r + 11$.
Nous simplifions aussi la notation des poids pour passer des neurones d'entrées aux neurones de la première couche cachée, en notant $\omega_{i,j}$ le poids pour passer du i^{eme} neurone d'entrée au j^{eme} neurone de la couche cachée.
Nous devons maintenant estimer tous ces paramètres, c'est à dire les
$(b, b_1, ..., b_{n_n}, \omega_{1,1}, ..., \omega_{n_x,n_n}, \Omega_1, ..., \Omega_{n_n})$.

Nous commençons par initialiser tous ces paramètres aléatoirement.
Ensuite pour chacune des lignes de notre base de données d'entraînement, constituée de n_x données d'entrée $(x_1, ..., x_{n_x})$ et une donnée réponse y_x, nous construisons la prédiction $\hat{y}_x$ en faisant passer les $(x_1, ..., x_{n_x})$ dans notre réseau.
Nous mettons à jour ces paramètres avec une idée simple.
Par exemple, pour les $\omega_{i,j}$ on calcule la quantité $\frac{\partial E}{\partial \omega_{i,j}}$, c'est à dire le sens, et l'amplitude de la modification de l'erreur de prédiction par une faible modification du $\omega_{i,j}$.

- Pour un $\frac{\partial E}{\partial \omega_{i,j}}$ relativement proche de zéro, il est inutile de trop modifier le paramètre $\omega_{i,j}$.
- Une valeur fortement négative de $\frac{\partial E}{\partial \omega_{i,j}}$ implique qu'une petite variation positive de $\omega_{i,j}$ implique une grande variation négative (donc souhaitable) de l'erreur.
- Une valeur fortement positive de $\frac{\partial E}{\partial \omega_{i,j}}$ implique qu'une petite variation positive de $\omega_{i,j}$ implique une grande variation positive (non souhaitable) de l'erreur.

Ainsi, on met à jour le paramètre $\omega_{i,j}$ avec :

$$\omega_{i,j} \leftarrow \omega_{i,j} - \eta \frac{\partial E}{\partial \omega_{i,j}}$$

avec $E = \frac{1}{2}(y_x - \hat{y}_x)$, et η le learning rate.
Ce paramètre η qui contrôle la vitesse d'apprentissage du réseau. Plus η est grand, plus vite le paramètre arrivera à un minimum, avec le risque cependant de tomber sur un minimum local et non global.
Ainsi, pour chaque x, nous commencerons par mettre à jour les $(b, \Omega_i, ...\Omega_{n_n})$, puis les $(b_1, ..., b_{n_n}, \omega_{1,1}, ..., \omega_{n_x,n_n})$

Nous avons maintenant besoin de calculer les dérivées partielles de l'erreur par rapport aux différents paramètres.
Rappelons que :

$E = \frac{1}{2}(y_x - \hat{y}_x)^2$
$\hat{y}_x = b + \sum_{i=1}^{n_n} \Omega_i h_i$
$h_i = f(b_i + \sum_{k=1}^{n_x} \omega_{k,i} x_k) = f(z_i)$
$f(x) = \frac{1}{1+e^{-x}}$

Par ailleurs, $\frac{\mathrm{d}f}{\mathrm{d}x} = \frac{e^{-x}}{1+\exp^{-x}} = f(x)(1 - f(x))$
Nous avons alors :

$$\frac{\partial E}{\partial b} = \frac{\partial E}{\partial \hat{y}_x} \frac{\partial \hat{y}_x}{\partial b}$$

$$= (y_x - \hat{y}_x)$$

$\forall i \in [\![1, n_n]\!]$

$$\frac{\partial E}{\partial \Omega_i} = \frac{\partial E}{\partial \hat{y}_x} \frac{\partial \hat{y}_x}{\partial \Omega_i}$$

$$= (y_x - \hat{y}_x) h_i$$

$\forall i \in [\![1, n_n]\!]$

$$\frac{\partial E}{\partial b_i} = \frac{\partial E}{\partial \hat{y}_x}\frac{\partial \hat{y}_x}{\partial h_i}\frac{\partial h_i}{\partial z_i}\frac{\partial z_i}{\partial b_i}$$

$$= (y_x - \hat{y}_x)\Omega_i f'(z_i)$$
$$= (y_x - \hat{y}_x)\Omega_i f(z_i)(1 - f(z_i))$$

$\forall i \in [\![1, n_n]\!], \forall k \in [\![1, n_x]\!]$

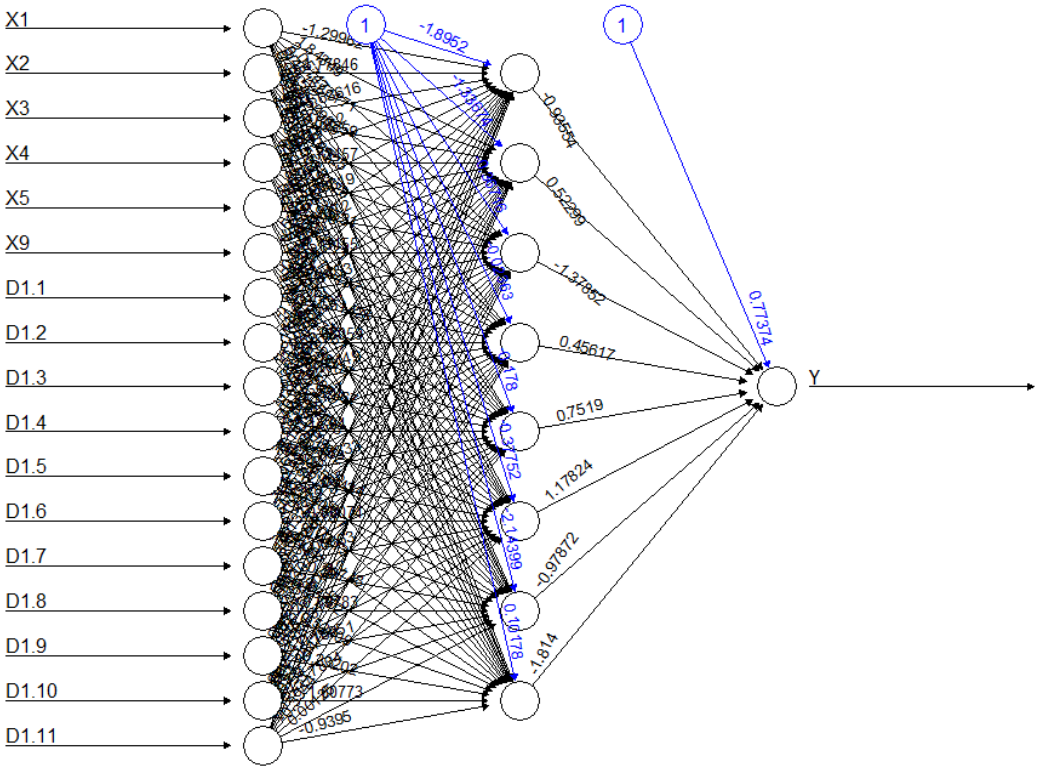
$$\frac{\partial E}{\partial \omega_{k,i}} = \frac{\partial E}{\partial \hat{y}_x}\frac{\partial \hat{y}_x}{\partial h_i}\frac{\partial h_i}{\partial z_i}\frac{\partial z_i}{\partial \omega_{k,i}}$$

$$= (y_x - \hat{y}_x)\Omega_i f(z_i)(1 - f(z_i))x_k$$

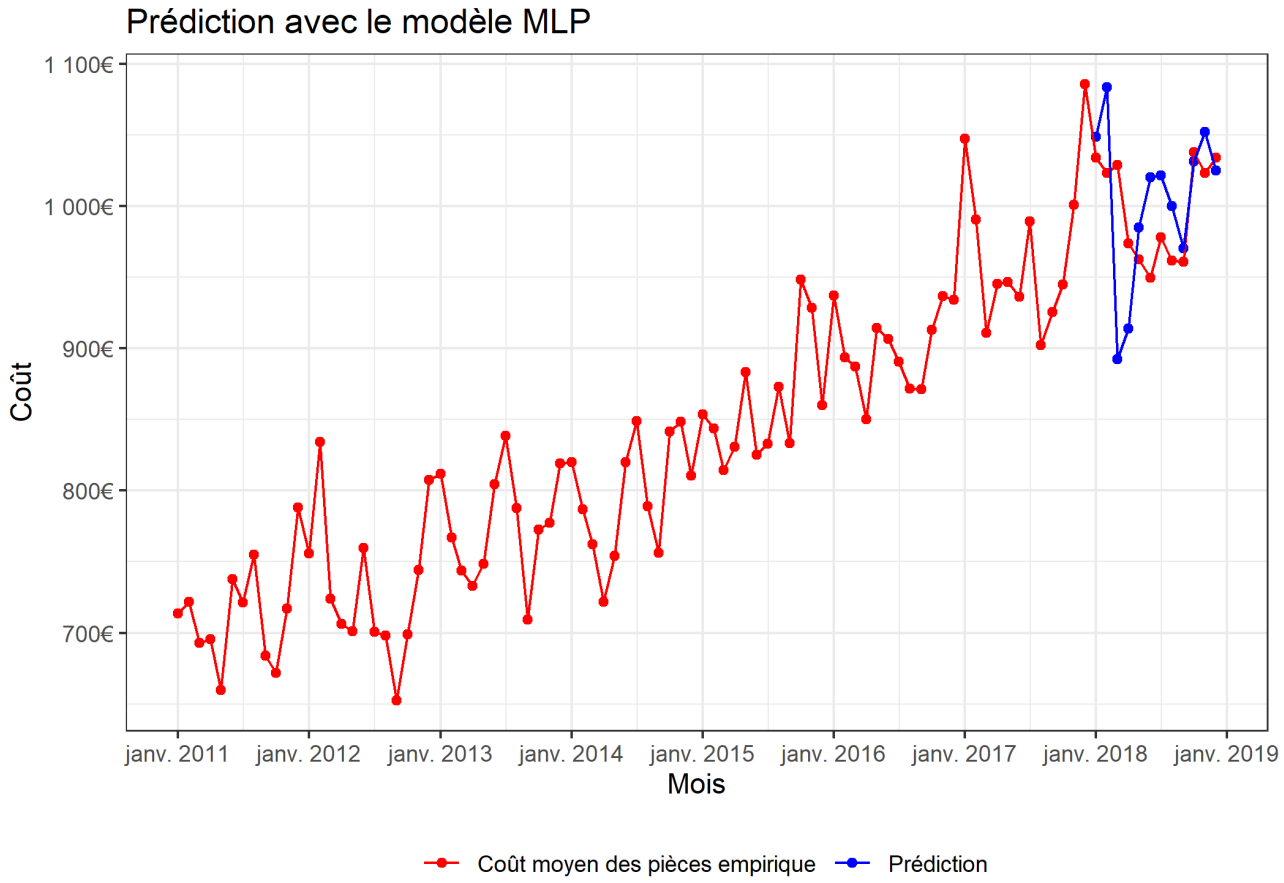
Nous appliquons ainsi ce procédé pour toutes les lignes de notre base de données d'entraînement.

4.5.4 Résultats du processus d'apprentissage

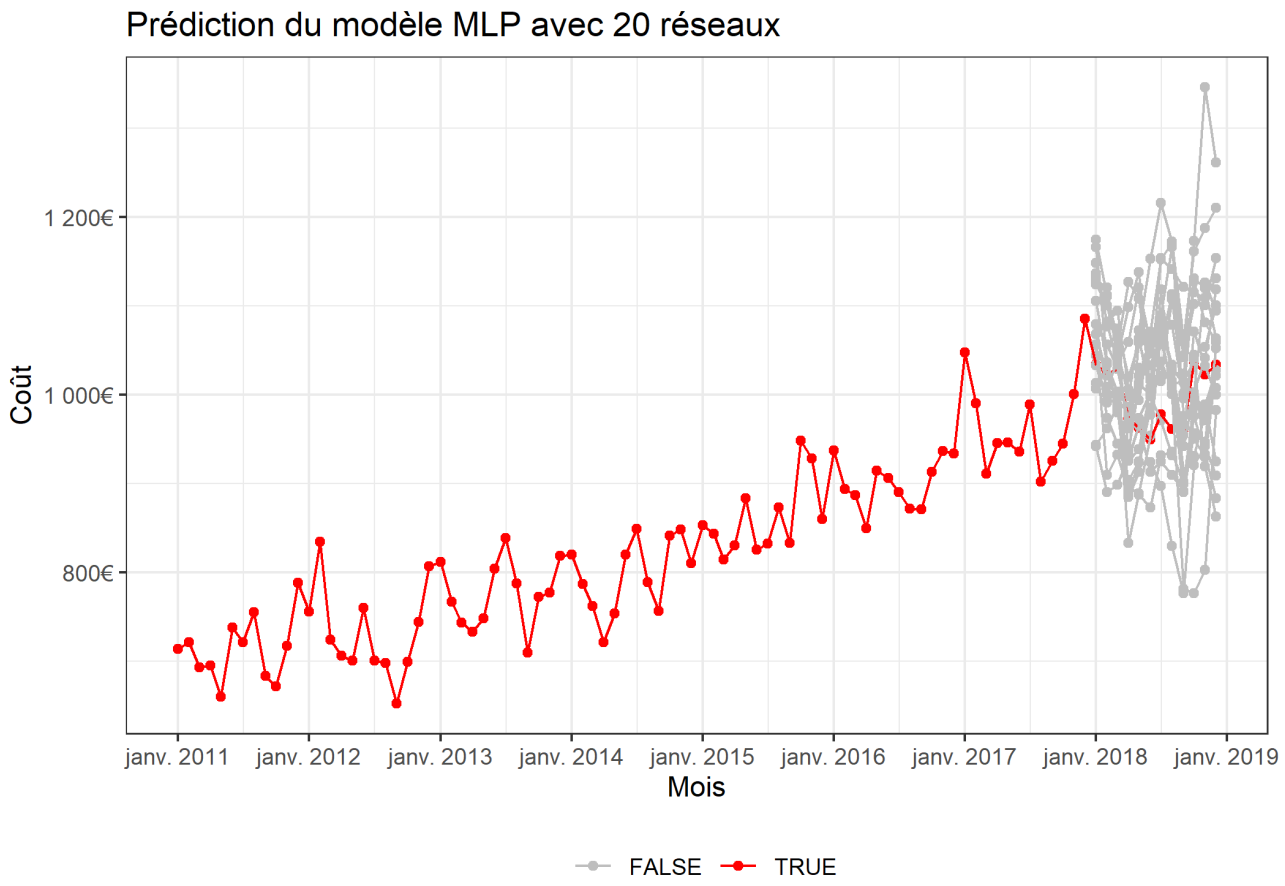
Nous construisons ainsi un réseau avec le processus décrit ci dessus. À la fin de ce processus d'apprentissage nous obtenons les coefficients suivants :



Nous construisons ainsi les prédictions sur les 12 prochains mois.



Cependant, comme nous l'avons dit précédemment, les prédictions sont aléatoires dues à l'initialisation aléatoire des coefficients de départ.
Ainsi, en construisant 20 réseaux différents, nous produisons 20 prédictions différentes soit, par exemple :

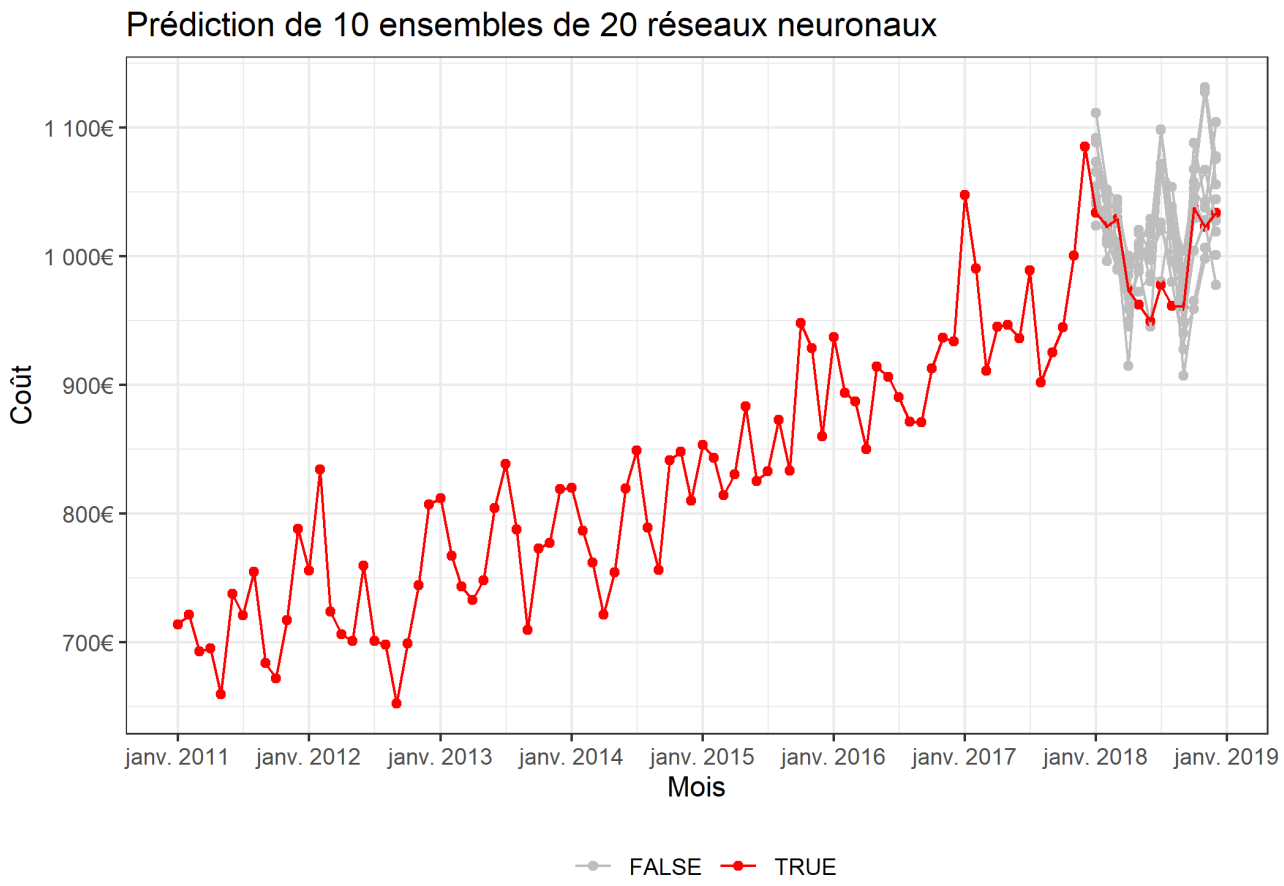


Nous ne pouvons donc pas nous fier sur la prédiction d'un seul réseau de neurones. Nous avons donc besoin de construire plusieurs réseaux de neurones, et nous construirons chaque prédiction $\hat{y}_{t+h}$ comme la médiane des prédictions de l'ensemble de tous les réseaux, soit :

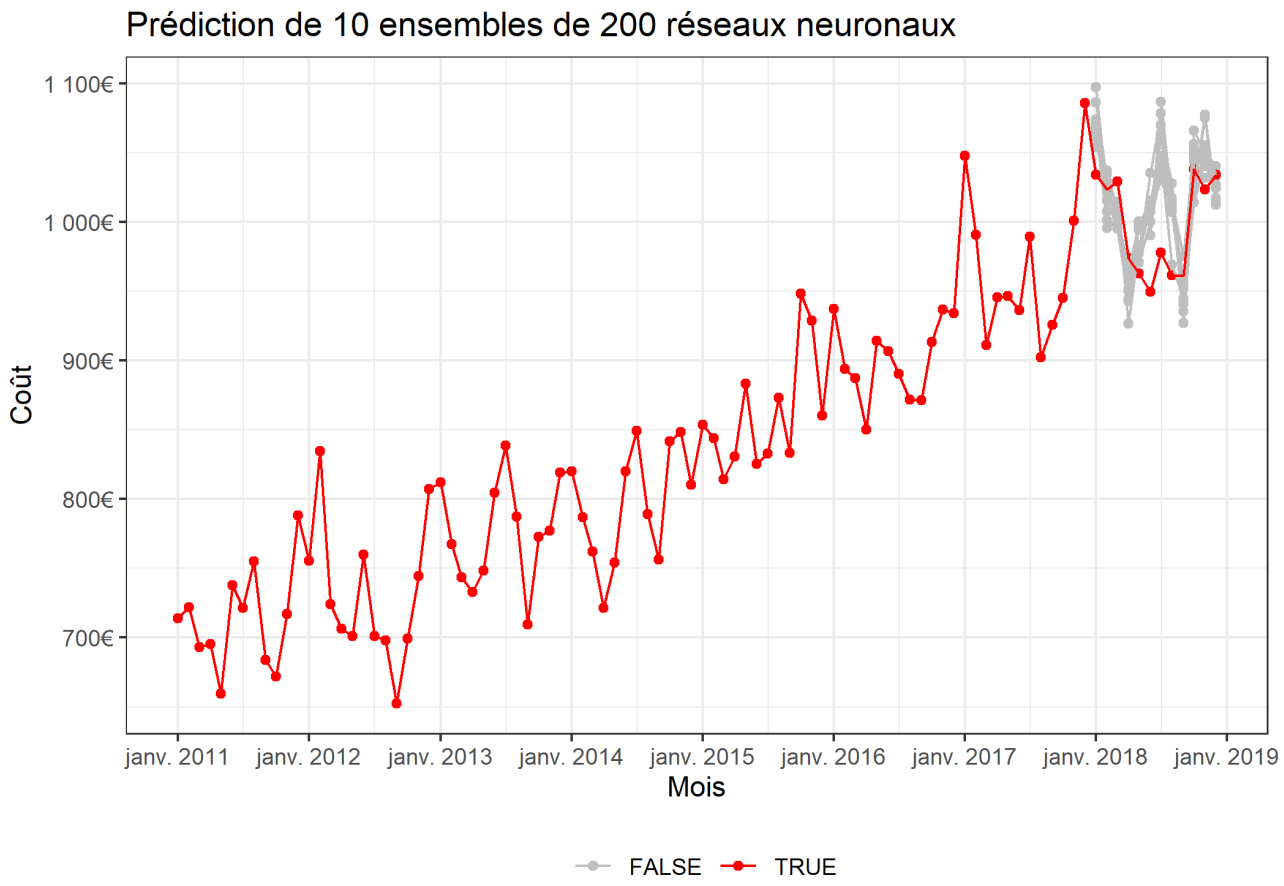
$$\forall h \in \mathbb{N} \quad \hat{y}_{t+h} = median(\hat{y}_{t+h}^{Rseau_1}, ..., \hat{y}_{t+h}^{Rseau_N})$$

Il reste maintenant à savoir le nombre de réseaux à construire pour avoir une prédiction stable.

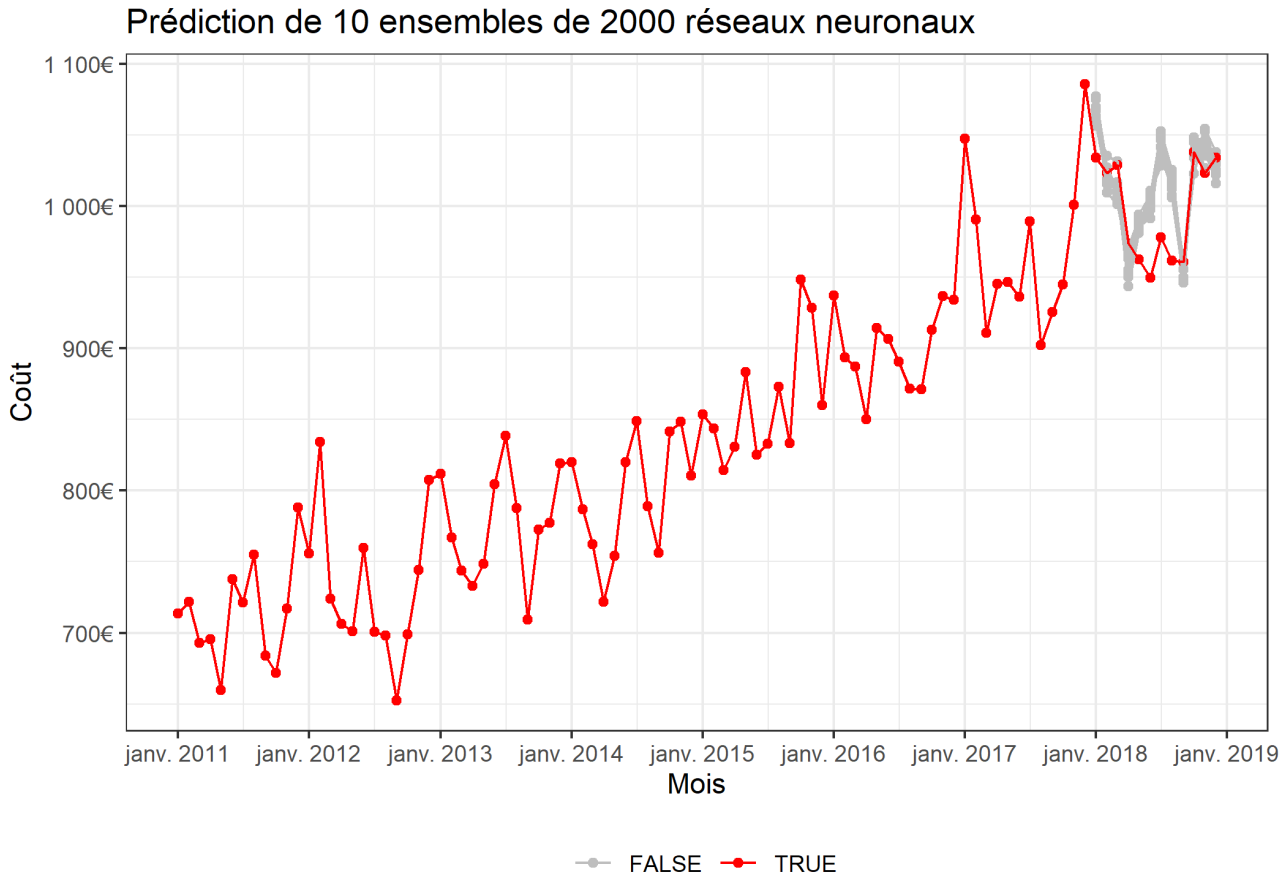
On a, pour 10 ensembles indépendants de 20 réseaux de neurones les prédictions suivantes :



20 réseaux de neurones ne semblent pas suffisants pour stabiliser la variance de nos prédictions.
Essayons avec 200 réseaux :

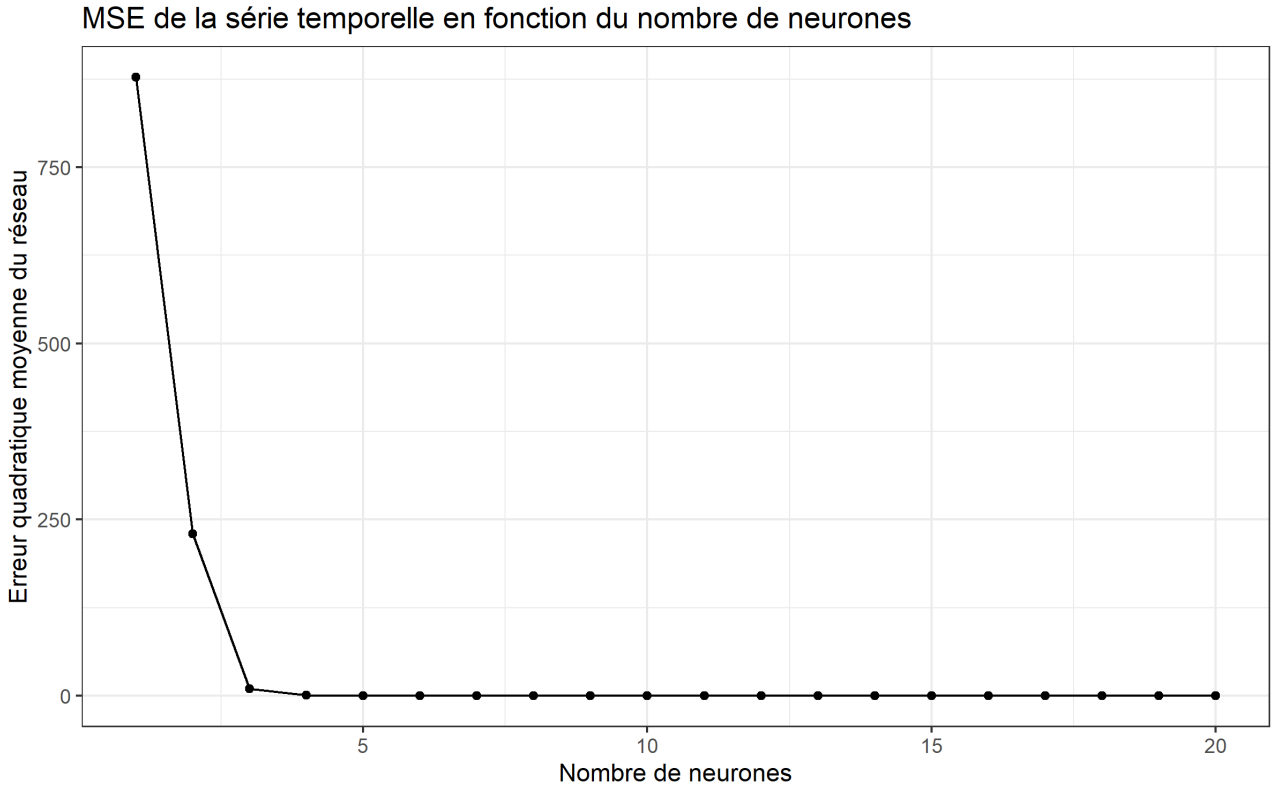


Ici, le résultat est beaucoup plus satisfaisant, avec des prédictions qui semble être stables. Regardons cependant le résultat avec 2000 réseaux de neurones :

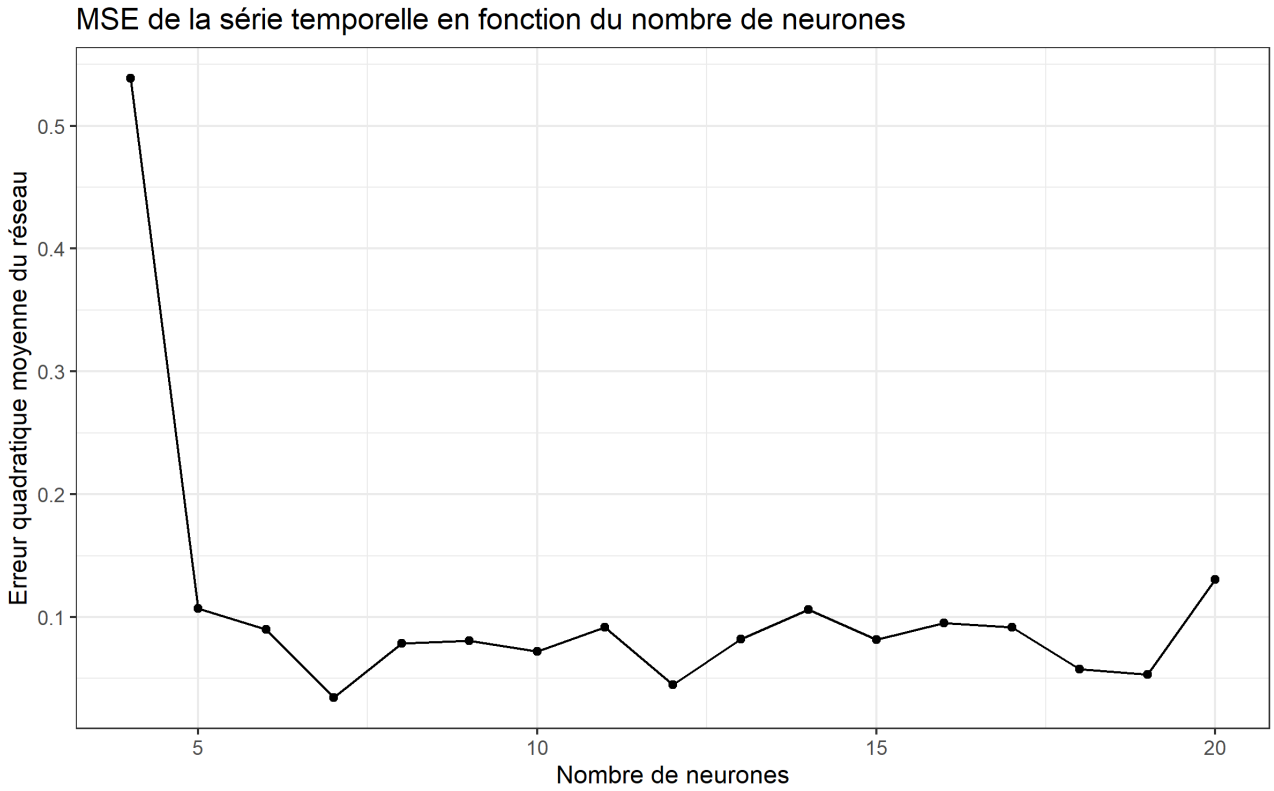


Le résultat avec 2000 neurones est semblable à celui à 200 neurones, tout en multipliant par 10 le temps de calcul.
C'est pourquoi, pour nos prédictions finales, nous utiliserons des ensembles de 200 neurones afin d'avoir des temps de calculs raisonnables.

Il reste maintenant à optimiser le nombre de neurones cachés. En effet, on a présenté pour l'instant les résultats pour 6 neurones cachés, cependant ce nombre peut être optimisé.
Pour cela , on va utiliser le MSE (Mean square error), c'est à dire l'erreur quadratique moyenne du réseaux de neurones. Pour chaque réseau entraînés, on repasse les données d'entraînement et on calcule l'erreur quadratique moyenne de ces données.
Ainsi, on va calculer le MSE de notre série temporelle test pour un nombre de neurones cachés allant de 1 à 20.
On obtient :

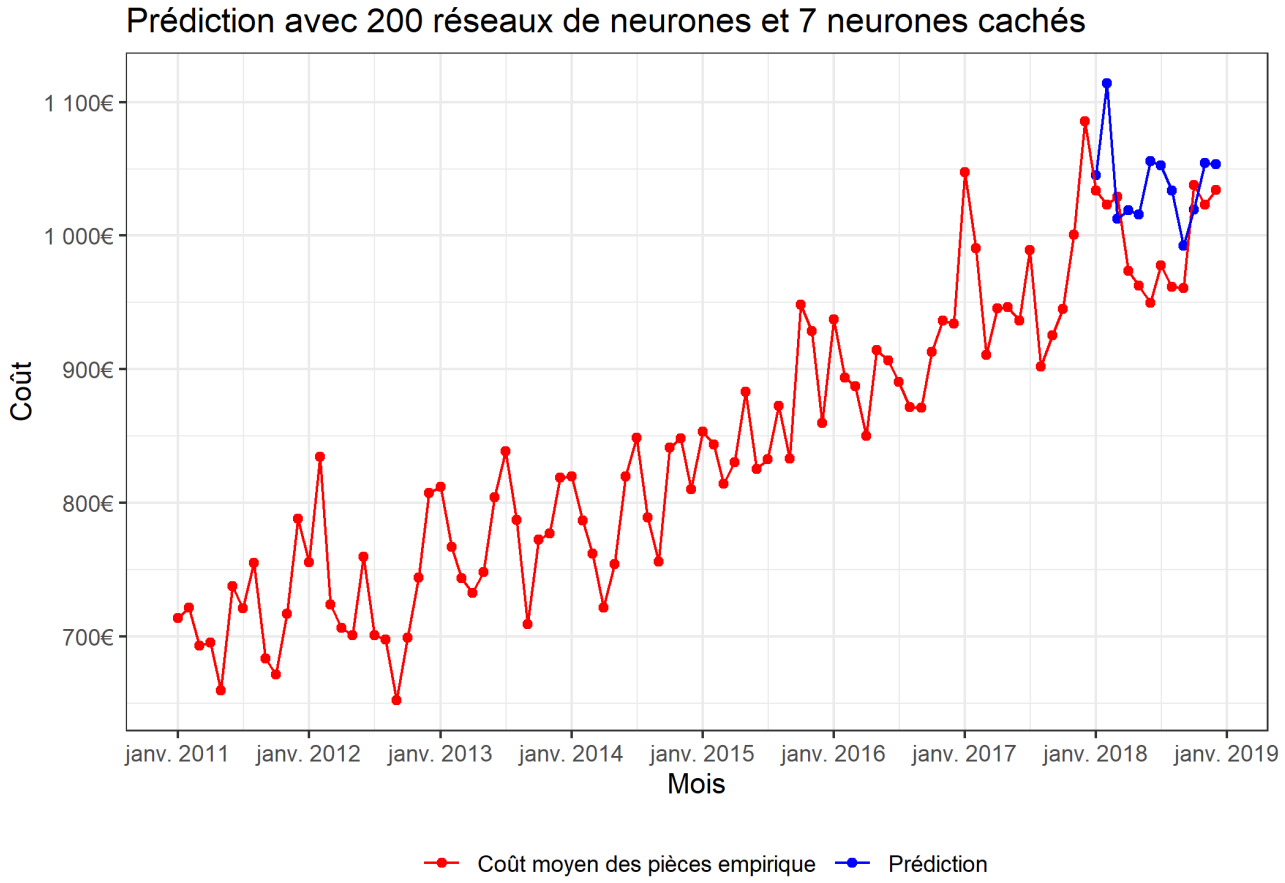


On voit que le MSE est très mauvais pour des modèles contenant 1 à 3 neurones cachés. Regardons de plus près ce qu'il se passe après 4 neurones cachés :



On voit ainsi que l'optimum est réalisé pour 7 neurones cachés.

Ainsi, dans toute la suite de ce mémoire, nous construirons nos modèles *mlp* avec 200 réseaux de neurones avec 7 neurones dans la couche cachée.
On a ainsi, au final, pour notre base de données, les résultats suivants :



On obtient une erreur moyenne en valeur absolue de 36.40€ et un pourcentage moyenne d'erreur en valeur absolue de 3.74% ce qui est moins bien que les modèles précédents.
Les modèles MLP présentent plusieurs désavantages :

— Prédictions présentant un facteur aléatoire.
— Temps d'entraînement des réseaux très long, surtout si on construit beaucoup de réseaux, ce qui est nécessaire pour avoir une prédiction stable.

Nous avons détaillé dans cette partie les aspects théoriques de 4 modèles de prédictions de séries temporelles, que ce soit des modèles simples et classiques (ARIMA,ETS), ou des modèles plus complexes et récents (TBATS et MLP).
Nous présenterons ci-dessous les résultats de ces différents modèles pour la prédiction du coût moyen des pièces à l'aide de nos variables explicatives présentées en deuxième partie.

4.6 Intervalle de prédictions

Au cours de notre exposé des différents modèles de prédictions de séries temporelles, nous avons volontairement omis les intervalles de prédictions.

Définition : Intervalle de prédiction Un intervalle de prédiction est une plage de valeurs qui est susceptible de contenir une observation individuelle future à partir des valeurs des prédicteurs en entrée, qui sont pris en compte dans un modèle.

En appliquant cette définition à la prévision de séries temporelles, il vient 5 types d'erreurs possibles :

1. Les erreurs individuelles aléatoires
2. Les erreurs sur l'estimation des paramètres
3. Les erreurs sur le nombre de paramètres du modèle
4. L'incertitude quand à la pertinence du modèle pour l'historique des données choisies
5. Si le point 4. est vérifié, est ce que le modèle continuera il d'être pertinent dans le futur ?

Un intervalle de confiance est une estimation de l'incertitude statistique des paramètres estimés dans le modèle. Il estime habituellement la source d'incertitude 2. conditionnellement à la 3., ne s'intéresse pas aux conditions 1.,4. et 5.
Un intervalle de prédiction devrait idéalement prendre en compte les cinq sources.
Malheureusement, les méthodes habituelles de prévision de séries temporelles ne tiennent généralement compte que de la source 1), les erreurs individuelles aléatoires. Cela diffère des intervalles de prédiction standards des modèles de régression plus simples et des modèles linéaires généralisés, qui tiennent compte, du moins en général, de l'incertitude des estimations des paramètres.
Le problème est que,pour toutes les méthodes de prévision de séries temporelles, il n'existe pas de moyen simple d'estimer l'incertitude qui découle de l'estimation des paramètres à partir des données. De même il n'est pas en général possible d'estimer l'incertitude des méta-paramètres comme la quantité de différenciation nécessaire, le nombre de termes auto-regressifs pour les modèles ARIMA etc...

Vérifions la précision de l'estimation des intervalles de confiances construits par les modèles de séries temporelles. Pour cela prenons un exemple tiré de nos données :
Nous allons prendre la série temporelle du coût moyen **journalier** des pièces (afin d'obtenir le plus de points possible).
Pour un modèle, en supposant que les erreurs sont distribuées selon une loi normale, nous avons :

$$y_{T+h|T} \sim \mathcal{N}(\hat{y}_{T+h|T}, \hat{\sigma}_h)$$

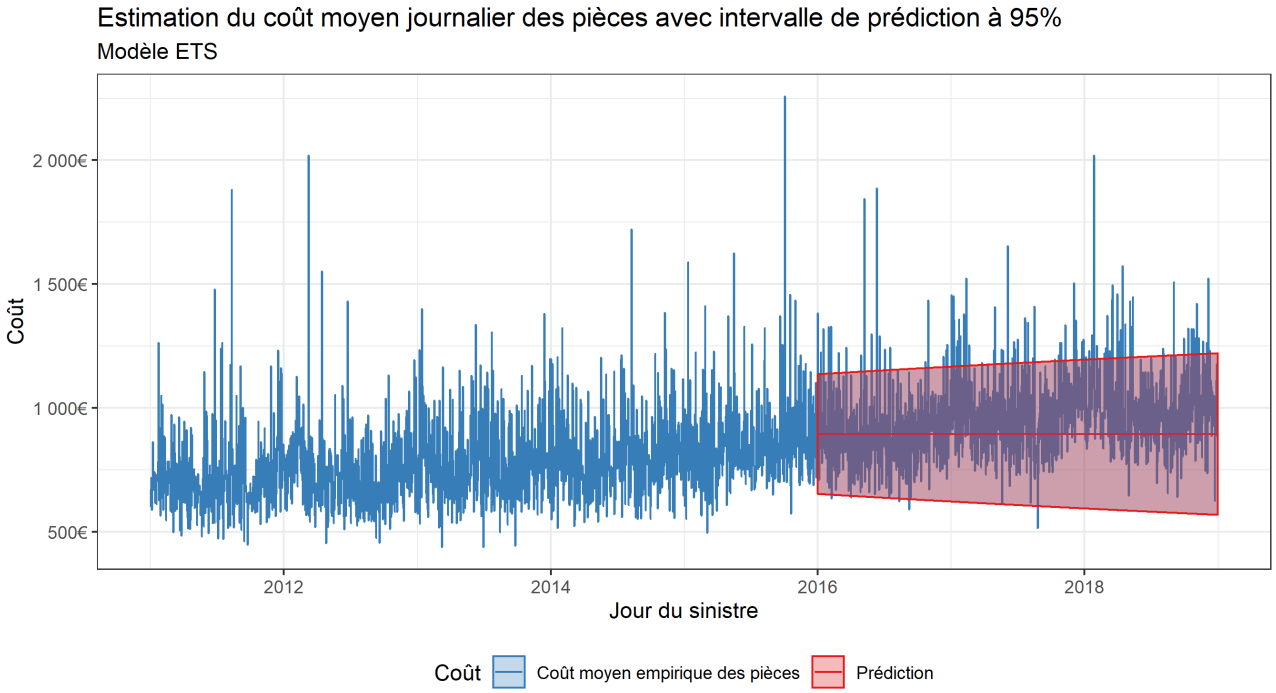
avec :

— $\hat{y}_{T+h|T}$ la prédiction effectuée par le modèle en question.
— $\hat{\sigma}_h$ l'écart type de l'erreur de prédiction en h. Classiquement, les modelès du package *forecast* prennent $\hat{\sigma}_h = \sigma\sqrt{h(1+\frac{h}{T})}$avec σ l'écart type des résidus.

Ainsi, l'intervalle de prédiction avec un niveau d'erreur de α, est de : $\hat{y}_{T+h|T} \pm q_{1-\frac{\alpha}{2}}\hat{\sigma}_h$ avec $q_{1-\frac{\alpha}{2}}$ le quantile de niveau $1-\frac{\alpha}{2}$ de la loi normale centrée réduite.

Regardons l'intervalle de prédiction déterminé par le package forecast pour notre série temporelle du coût moyen journalier des pièces. Nous couperons cette série temporelle au 1er janvier 2016 pour constituer les séries temporelles d'entraînement et de test. Nous tracerons l'intervalle de prédiction à 95%.

Avec le modèle ETS, nous obtenons :



Nous obtenons ainsi 101 point en dehors de l'intervalle de confiance sur 1096 points au total, soit un taux d'erreurs de 9.2%.
Nous voyons donc que l'estimation de l'intervalle de prédiction est assez mauvais. Ceci est dû au fait que le modèle est mal adapté aux données.

Ainsi, à cause du manque de fiabilité que l'on peut avoir dans les intervalles de prédiction fournis par nos fonctions R, nous ne produirons pas d'intervalle de prédictions pour nos modèles. En plus des erreurs sur les intervalles de prédiction ci dessous, d'autres types d'erreurs dus à notre méthodologie vont émerger. En effet, nous serons amené à diviser notre base d'entraînements selon une ou plusieurs variables, pour ensuite recomposer le coût moyen global à l'aide de l'estimation empirique de la répartition des sinistres. Cette répartition des sinistres est elle aussi une source d'incertitude difficile à quantifier.

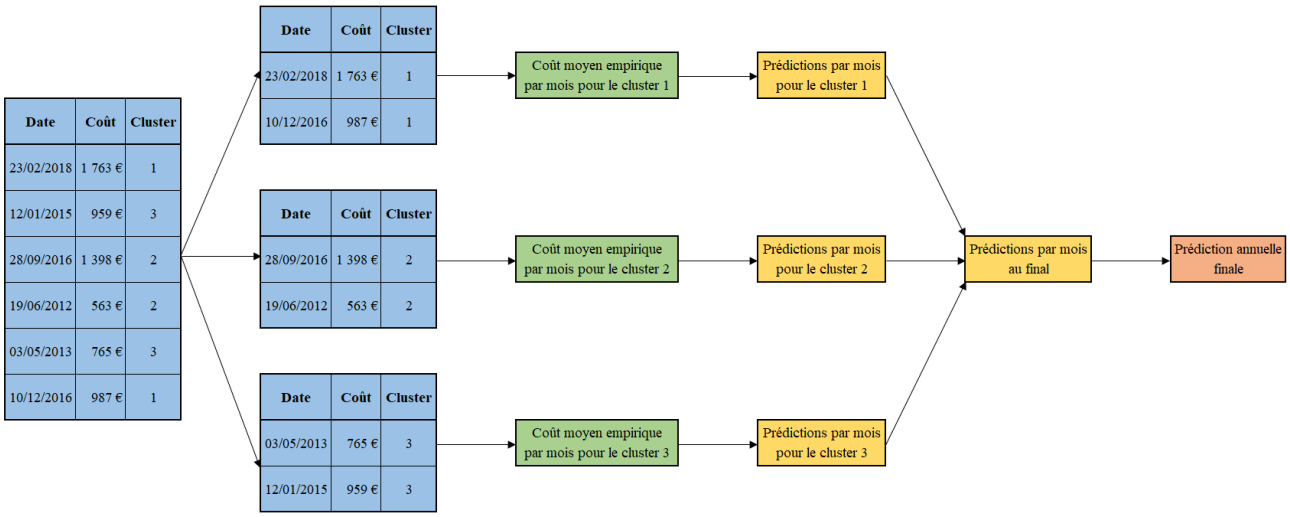
Chapitre 5

Détermination de la prédiction finale

5.1 Rappel sur la méthode générale

Dans les deux chapitres précédents, nous avons en premier lieu construit des variables explicatives, ligne à ligne pour chaque sinistres. Ces variables explicatives sont au nombre de 13, et comportent chacune un faible nombre de modalités (au maximum 6).
Puis nous avons vu dans le chapitre précèdent 4 modèles de séries temporelles, qui, à partir d'une série temporelle, renvoie une prédiction pour un horizon choisi. Nous avons déjà construit des prédictions en utilisant la série temporelle du coût moyen des pièces par mois.
Cependant, afin d'affiner les prédictions, nous allons maintenant utiliser les variables explicatives construites en partie 2

Prenons l'exemple de la variable explicative "Cluster Kilométrage", dont nous avons expliqué la construction ici.
Cette variable comporte trois modalités ("1","2","3"). On va ainsi diviser la base de données d'entraînement en trois bases plus petites, selon les modalités "1","2","3".(Partie bleu du schéma ci dessous)
On calcule ensuite, pour chacune des 3 bases de données le coût moyen des pièces par mois,afin de constituer nos 3 séries temporelles. (Partie verte du schéma)
Ensuite, pour chacune de ces 3 séries temporelles, on choisit le modèle souhaité (ARIMA, ETS, TBATS, MLP), puis on établit des prédictions. (Partie jaune du shéma)
Ensuite, nous reconstituons la prédiction finale en effectuant une somme pondérée des trois prédictions précédentes. Les poids de cette somme sont la répartition empirique des sinistres suivant les modalités en question. (Si la modalité "1" représente 30% du nombre de sinistres, la modalité "2" 50% du nombre de sinistres, et la modalité "3" 20% du nombre de sinistres, les poids de la somme pondérée seront 0.3,0.5,0.2).
Finalement, nous obtenons la prédiction annuelle en effectuant une somme pondérée des coûts moyens prédits chaque mois. Les poids de pondération correspondent à la répartition des sinistres pour chaque mois de l'année.



5.2 Méthode pour déterminer la pertinence d'un modèle

Le but principal de cette étude est de déterminer le coût moyen des pièces pour l'année $N+1$ à vision de l'année N.
Pour tester la pertinence de chaque prédiction, deux choix s'offrent à nous :

- Étudier l'erreur du modèle pour la prédiction du coût moyen des pièces de manière annuelle.
- Étudier l'erreur du modèle pour la prédiction du coût moyen des pièces de manière mensuelle.

Comme le but principal est d'estimer le coût moyen annuel des pièces, la première méthode semble être la meilleure, cependant nous ne disposons pas de suffisamment d'années de tests pour que cette information soit pertinente.

Nous allons ainsi tester nos modèles par rapport à leur capacité à bien estimer le coût moyen **mensuel** des pièces.
Ainsi, nous construirons nos modèles sur notre base de données de 2011 à 2015 pour les tester sur les mois de l'année 2016, puis nous reconstruirons nos modèles de 2011 à 2016 pour les tester sur l'année 2017 pour finalement reconstruire nos modèles de 2011 à 2017 pour les tester sur l'année 2018.
Nous séparerons nos bases de données selon 2 variables maximum. En effet au delà de deux variables, nos prédictions se dégradent tout en alourdissant les temps de calculs.

Ainsi, nous construirons chaque modèle, séparés par chaque groupes de variables possible 3 fois.

- La première pour construire nos modèles sur une base de données comprenant les sinistres survenus entre 2011 et 2015, pour ensuite tester ce modèle sur les mois de 2016.
- La deuxième pour construire nos modèles sur une base de données comprenant les sinistres survenus entre 2011 et 2016, pour ensuite tester ce modèle sur les mois de 2017.
- La troisième pour construire nos modèles sur une base de données comprenant les sinistres survenus entre 2011 et 2017, pour ensuite tester ce modèle sur les mois de 2018.

Nous allons ainsi juger chaque modèle suivant sa performance sur 36 mois en tout.
Afin de juger la qualité de chaque prédictions, il s'agit de comparer les valeurs observées et les valeurs prédites par nos modèles à travers différentes fonctions de perte.

Dans la suite, nous utiliserons trois fonctions de perte classique :

—
$$MAE(\hat{y}) = \frac{1}{N}\sum_{i=1}^{N} \mid \hat{y}_i - y_i \mid \quad \textit{(Mean Absolute Error)}$$

—
$$RMSE(\hat{y}) = \frac{1}{N}\sum_{i=1}^{N} \sqrt{(\hat{y}_i - y_i)^2} \quad \textit{(Root-Mean-Square Error)}$$

—
$$ME(\hat{y}) = \sup_{[\![1,N]\!]} \mid \hat{y}_i - y_i \mid \quad \textit{(Maximum Error)}$$

Parmi ces trois fonctions de perte, nous privilégierons la première car elle correspond plus à ce que nous voulons prédire mais nous resterons cependant attentif aux performances de nos modèles suivant les deux autres fonctions de perte.

Nous avons donc 4 modèles, avec , en tout, 91 variables ou groupements de variables (plus la variable "Total" qui ne sépare pas la base de données).
Cependant,certains de nos modèles n'ont pas convergé pour certains regroupements de variables, c'est pourquoi les regroupements de variables ne sont pas présents.
Pour chaque prédiction, on calcule nos 3 fonctions de perte.
On pre´sente ci dessous les meilleurs prédictions, celles avec une MAE infe´rieure a` 30, classe´es par MAE croissantes.

Modele	Variable	MAE	RMSE	ME
TBATS	ClusterPUISSANCE_ADMINISTRATIVE_1-ClusterKILOMETRAGE_1	26.70810	35.79295	115.79533
ETS	GENREVEHICULE-ENERGIE_SRABIS	26.78653	36.31619	110.83804
TBATS	ENERGIE_SRABIS-ClusterKILOMETRAGE_1	27.56126	36.81385	103.80184
ETS	ENERGIE_SRABIS-ClusterKILOMETRAGE_1	27.65788	38.44097	103.47391
ETS	ENERGIE_SRABIS-ClusterPUISSANCE_ADMINISTRATIVE_1	27.77598	35.36924	97.48223
ETS	ENERGIE_SRABIS	28.07034	36.37918	95.52799
TBATS	TopAgree-ClusterPUISSANCE_ADMINISTRATIVE_1	28.43758	38.08057	118.77609
ETS	ENERGIE_SRABIS-ClusterNature_sinistre_1	28.50471	35.96410	95.85977
TBATS	Total	28.82636	37.30131	100.45526
TBATS	ClusterNB_CYLINDRE_1	29.12214	35.50489	93.87955
TBATS	ClusterCLAS_REPAR_1	29.19880	39.59384	103.40174
TBATS	ClusterAgeVehicule_2-ClusterPUISSANCE_ADMINISTRATIVE_1	29.54054	39.41961	114.11463
TBATS	GENREVEHICULE-ClusterKILOMETRAGE_1	29.60121	38.32462	106.06262
ARIMA	GENREVEHICULE-ClusterNature_sinistre_1	29.67063	41.86823	134.90161
ARIMA	ClusterPUISSANCE_ADMINISTRATIVE_1-ClusterNB_CYLINDRE_1	29.71252	41.07866	135.42117
TBATS	ClusterCLASDOM_SRA_ORIG_1	29.73404	39.15852	114.84355
ARIMA	ClusterNB_CYLINDRE_1	29.75520	40.19546	125.63306
MLP	ClusterMarque_2-ClusterCLASDOM_SRA_ORIG_1	29.83570	45.02077	147.28132
ARIMA	GENREVEHICULE-ClusterPUISSANCE_ADMINISTRATIVE_1	29.94014	42.33725	138.07615

On voit ainsi que les modèles TBATS et ETS semblent produire les meilleurs prédictions.
Nous vérifierons cette proposition ci dessous.

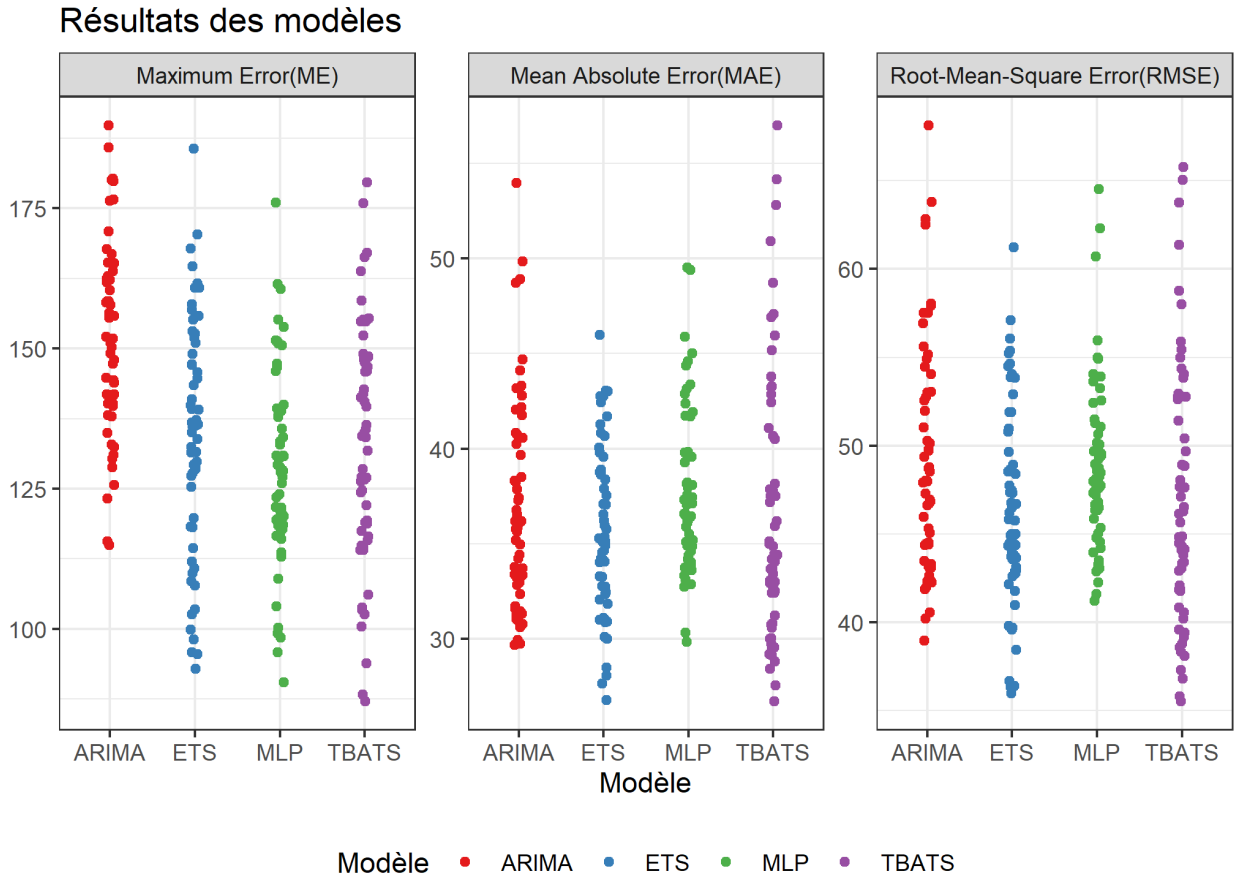
5.3 Détermination des meilleurs modèles

Comme évoqué précédemment, pour certains regroupements de variables, certains modèles ne convergent pas.
Ainsi,nous ne disposons pas de prédictions pour tous les regroupements de variables.
Nous obtenons :

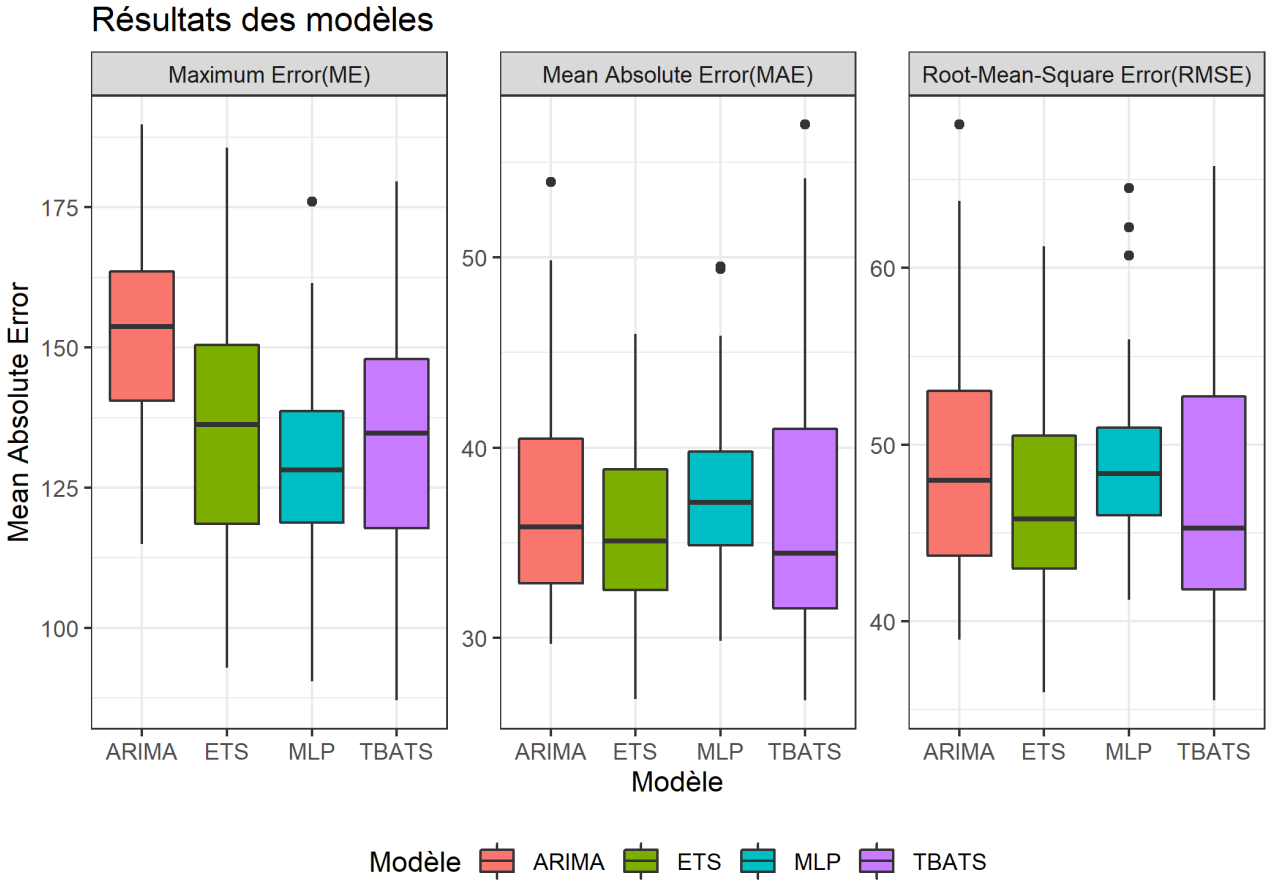
Modèles	Nombre de modèles qui ont convergé
MLP	58
TBATS	59
ARIMA	90
ETS	92

Nous ne prendrons ainsi en compte que les prédictions comprenant des regroupements de variables qui ont convergés pour les 4 modèles de séries temporelles.

Nous obtenons ainsi 58 regroupements de variables pour 4 modèles, soit 232 couples modèles/variables.
Nous regroupons les résultats au travers de nos 3 fonctions de perte, en les présentant par modèles.



Cela confirme que les meilleures prédictions sont produites par les modèles ETS et TBATS. Cependant, les modèles TBATS produisent des prédictions avec des MAE présentant une forte variance.
Le modèle MLP lui tend à sous performer selon la métrique MAE comparé aux modèles ETS et TBATS, mais performe mieux que ces modèles pour la métrique ME. Ainsi, les prédictions construites avec le modèle MLP n'ont pas d'erreurs importantes, mais ont une erreur moyenne plus grande que les autres modèles.
Le modèle ARIMA semble lui , ne pas avoir d'avantages par rapport aux 3 autres modèles.
Pour confirmer ces analyses, on trace les boites à moustaches (Boxplot) de ces même fonctions de perte des prédictions, pour chacun des modèles.
On obtient :

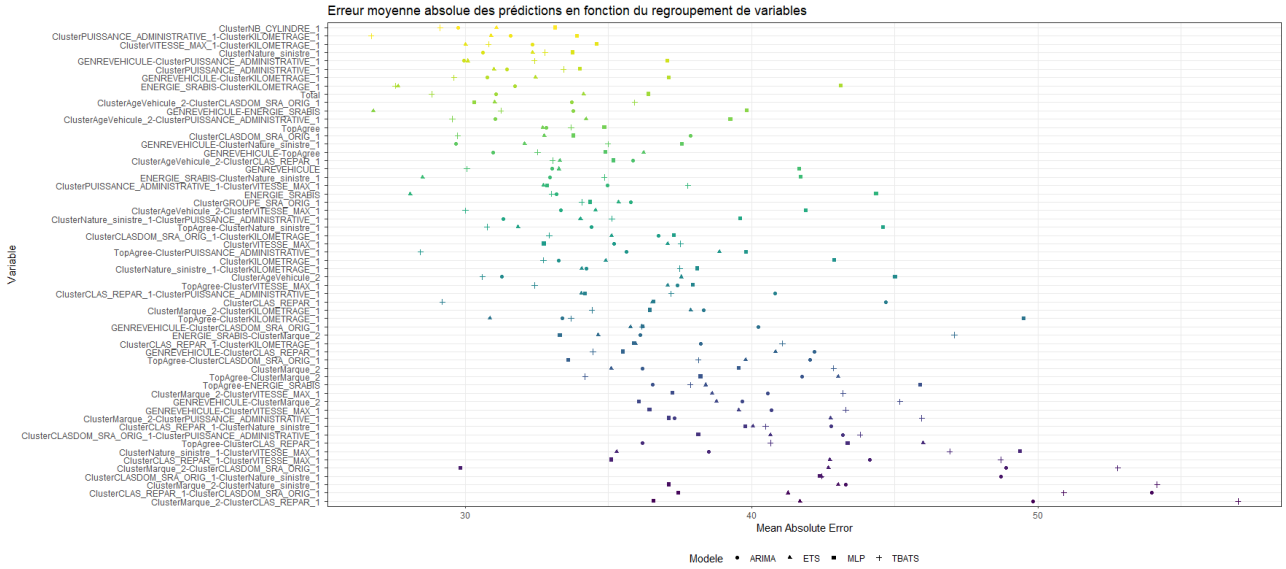


Nous voyons ainsi que, concernant la métrique MAE, le modèle $TBATS$ produit plus de 50% de bons modèles (les prédictions avec le modèle $TBATS$ ont la médiane de MAE la plus faible) mais le reste des modèles est très mauvais (les prédictions avec le modèle $TBATS$ ont l'écart inter quartile le plus fort, et les MAE les plus fortes). Ainsi, la modélisation complexe de la saisonnalité avec les modèles $TBATS$ est bénéfique pour certains regroupements de variables pour produire de meilleurs prédictions que les modèles ETS, mais induit aussi beaucoup de mauvaises prédictions pour d'autres regroupements de variables.

Les prédictions avec le modèle MLP produisent l'écart inter-quartile le plus faible pour la métrique MAE mais aussi la médiane la plus forte. Ainsi aucune des ces prédictions n'est catastrophiques ou aberrantes, mais aucune non plus n'est spécialement bonne.
Concernant l'erreur maximale (ME) le modèle MLP est le modèle qui produit les erreurs maximums les plus faibles.

5.4 Analyse des regroupements de variables

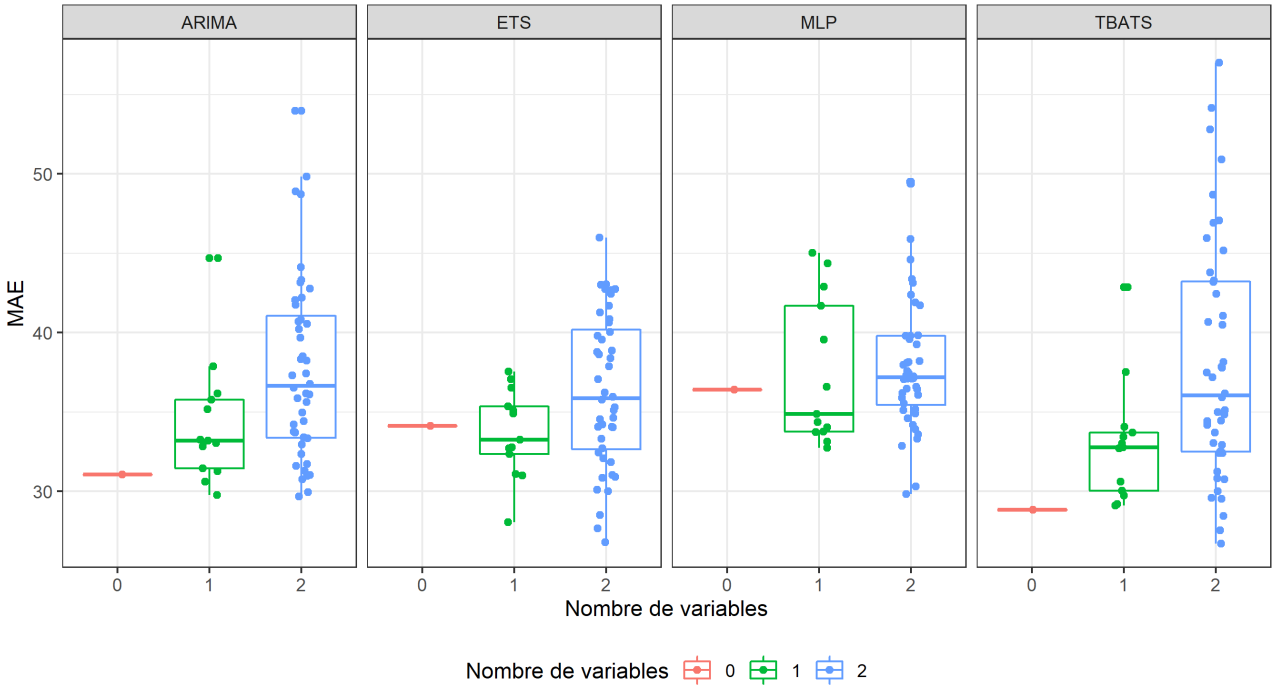
Regardons maintenant les résultats suivant les regroupements de variables.
Nous prenons ainsi les mêmes résultats que ceux ci-dessus, mais présentés de manière différentes.
Pour chacun des 58 regroupements de variables, nous regardons l'erreur moyenne absolue (MAE), en précisant le modèle avec la forme de chaque point. Nous obtenons :



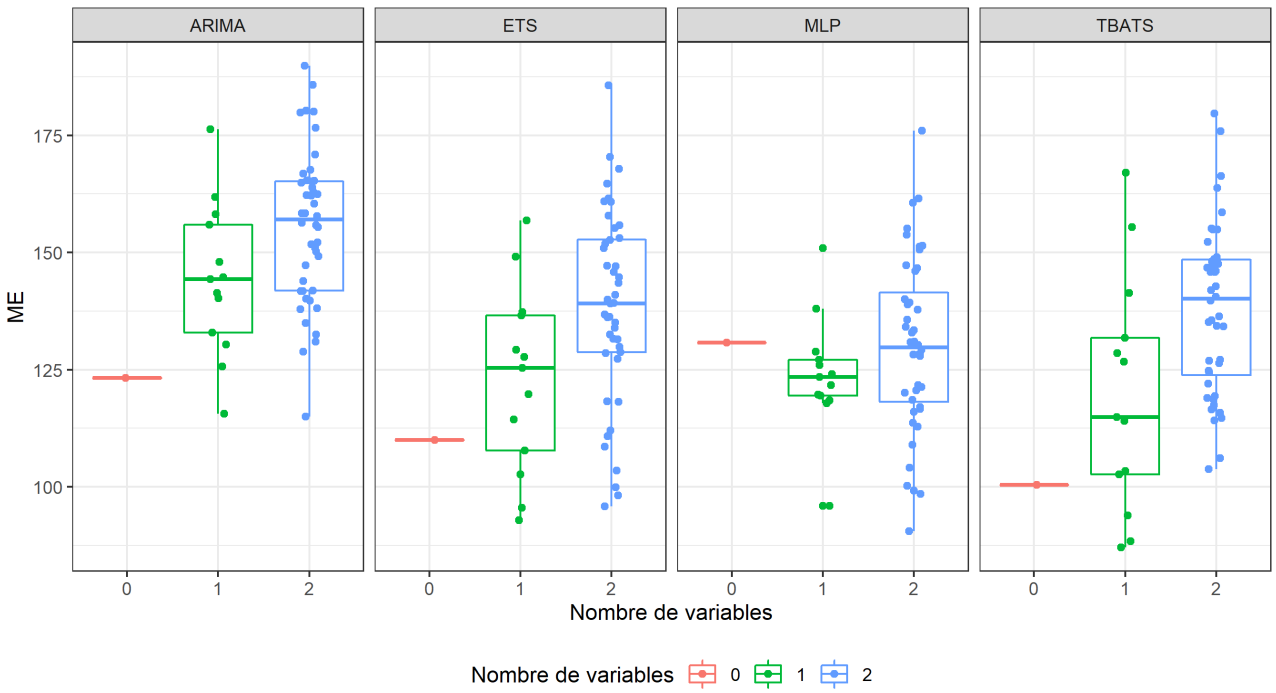
Les regroupements de variables sont classés ci dessus par ordre croissant de la moyenne des 4 erreurs moyennes absolues.
Nous voyons ainsi que beaucoup de regroupements de variables "mauvais" (c'est à dire les regroupements de variables présents en bas du graphiques) sont très souvent des regroupements à 2 variables.
Ceux ci sont aussi présents en haut du graphique, mais avec tout de même quelques regroupements à une variable.
Il est à noter que le regroupement "total", c'est à dire la prédiction construite en ne divisant pas la base d'entraînements en plusieurs plus petite, se classe en 9ème position des meilleurs regroupements de variable selon la moyenne des MAE.
Afin de confirmer ces affirmations, étudions les résultats de nos modèles en fonction du nombre de variables utilisée pour le regroupement.

5.5 Analyse sur les résultats en fonction du nombre de variables utilisées pour le regroupement

Nous affichons la boite à moustache avec les points correspondant des erreurs moyennes absolues de nos prédictions en fonction du nombre de variables utilisées pour le regroupement.
Nous obtenons :



Seuls les modèles MLP et ETS ont une médiane de MAE stable suivant le nombre de variables utilisé pour diviser la base de données En particulier, les divisions de base de données suivant une seule variable donnent de meilleurs prédictions en moyenne que si on ne divise pas la base de données. Pour les modèles ARIMA et TBATS, les prédictions ne bénéficient pas, en moyenne, de la division de la base de données.
Regardons si ce phénomène se vérifie également pour l'erreur maximum.



On voit ici que le modèle MLP bénéficie de la séparation de la base de données pour l'erreur maximum de la prédiction, et encore une fois la séparation suivant une variable est la

meilleure en moyenne pour cette métrique.
C'est cependant le seul modèle à bénéficier en moyenne de cette séparation pour l'erreur maximum.

On peut conclure ici que le modèle TBATS est, en tout point, le modèle qui bénéficie le moins de la séparation de la base de données ; alors que le modèle MLP est le modèle qui bénéficie le plus de ces séparations, spécialement lorsque l'on sépare la base de données suivant une variable.

5.6 Détermination de la prédiction finale

Afin de déterminer le meilleur couple modèle-variable, nous allons utiliser les conclusions des sections ci-dessus.
Le premier couple minimisant la fonction de perte MAE est la prédiction avec le modèle TBATS et le regroupement de variables
"Cluster Puissance administrative/Cluster Kilométrage". Ce regroupement de variables obtient, avec les 4 modèles, la deuxième meilleur moyenne de MAE.Ce regroupement est fiable car il semble aussi fournir de bonnes prédictions pour les autres modèles.
Par ailleurs l'erreur quadratique moyenne (RMSE) de ce couple est le troisième meilleure de nos prédictions. Mais l'erreur maximale se classe 29ème.
Nous allons cependant garder ce couple pour la prédiction.

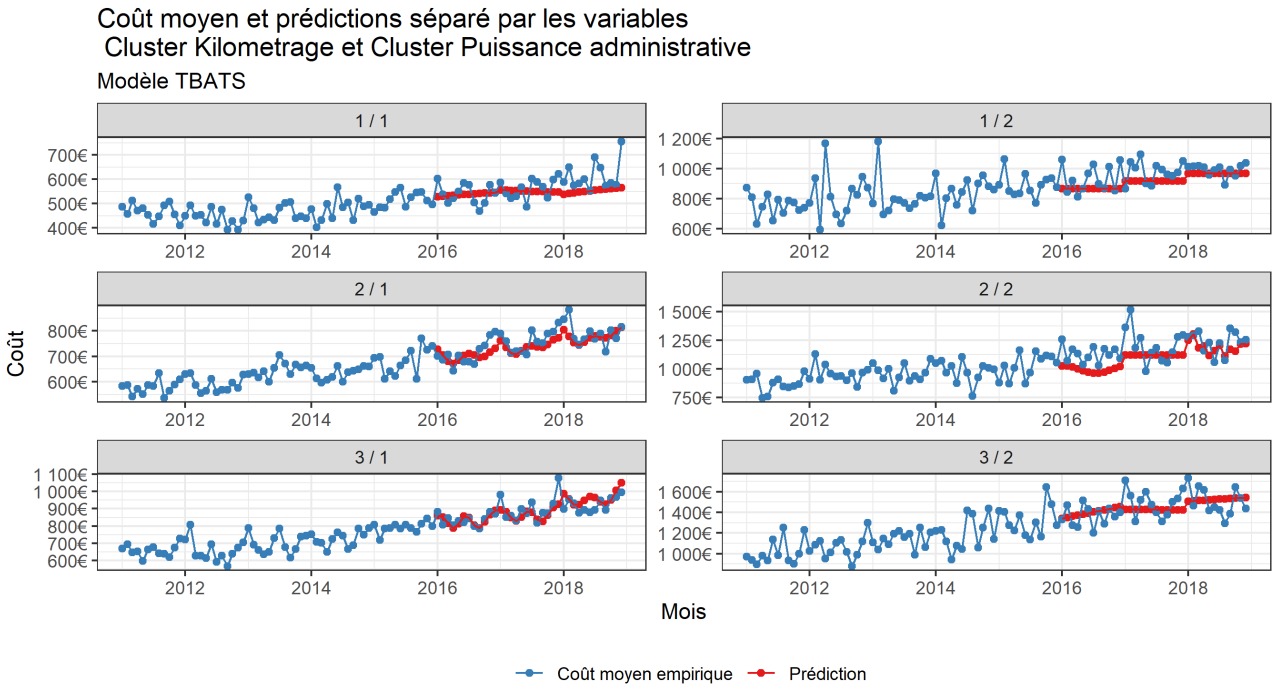
Le deuxième meilleur couple en terme de MAE est le couple ETS-Genre véhicule/ENERGIE_SRABIS. Or ce regroupement de variables performe mal pour les autres modèles (la moyenne des 4 MAE pour ce regroupement de variables se classe onzième). Nous ne prendrons donc pas en compte ce modèle.

Nous avons les mêmes problèmes avec les couples
TBATS-ENERGIE_SRABIS/Cluster Kilométrage et
ETS-ENERGIE_SRABIS/Cluster Kilométrage.

On prendra ainsi comme autre duo le couple ETS-ENERGIE_SRABIS qui se classe 5ème meilleur couple pour les métriques MAE,RMSE,et ME.
Regardons en détail les prédictions de ces deux couples :

TBATS-Cluster Puissance administrative/Cluster Kilométrage
Regardons la prédiction produite par le modèle TBATS en divisant la base de données selon deux variables :

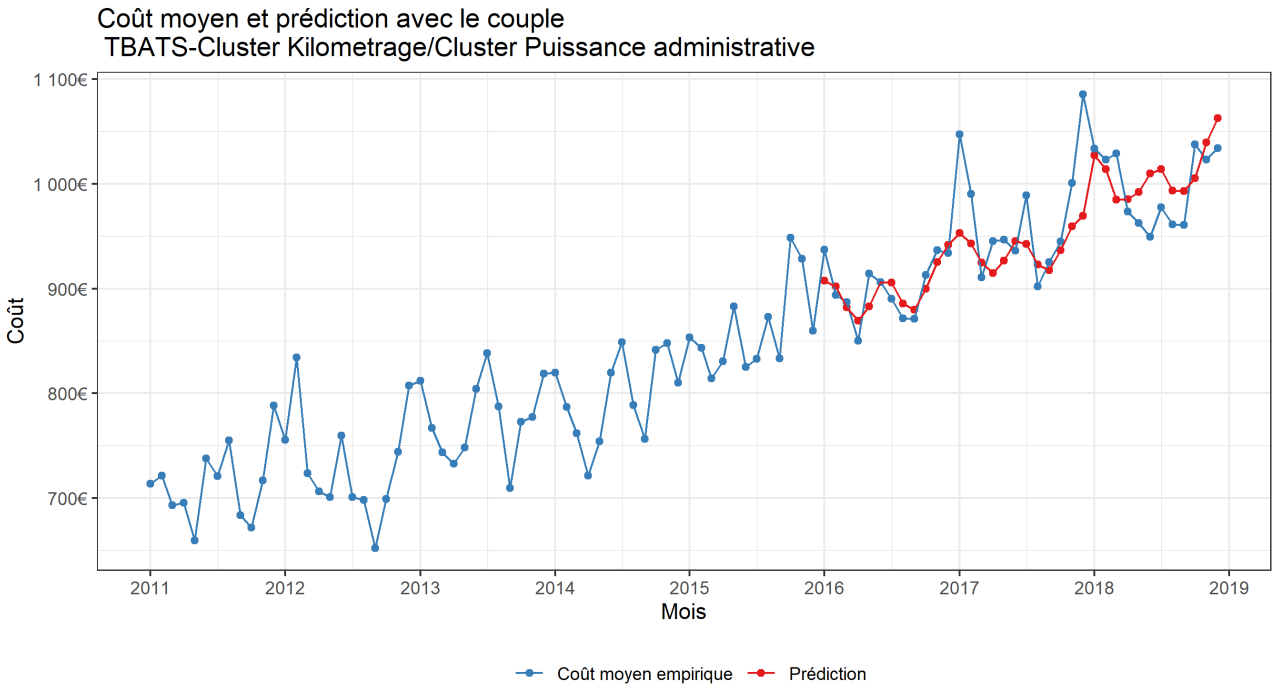
— La variable Cluster Puissance administrative comportant deux modalités
— La variable Cluster Kilométrage comportant trois modalités
On obtient ainsi, comme prédiction pour les années 2016, 2017 et 2018.



On voit ainsi que certaines prédictions sont très mauvaises, comme par exemple pour la série temporelle "1/1" et "1/2". Cependant, ces erreurs de prédictions sont atténuées par le fait que ces regroupements ont un faible nombre de sinistres, ainsi ce coût moyen aura un poids faible dans le coût moyen total.
Regardons en détail la répartition du nombre de sinistres entre nos 6 bases de données :

Cluster Puissance administrative	Cluster Kilométrage	Répartition
1	1	7.55%
	2	4.78%
	3	17.87%
2	1	9.17%
	2	44.64%
	3	15.99%

On peut ainsi reconstituer la prédiction pour le coût moyen total des pièces. On obtient :



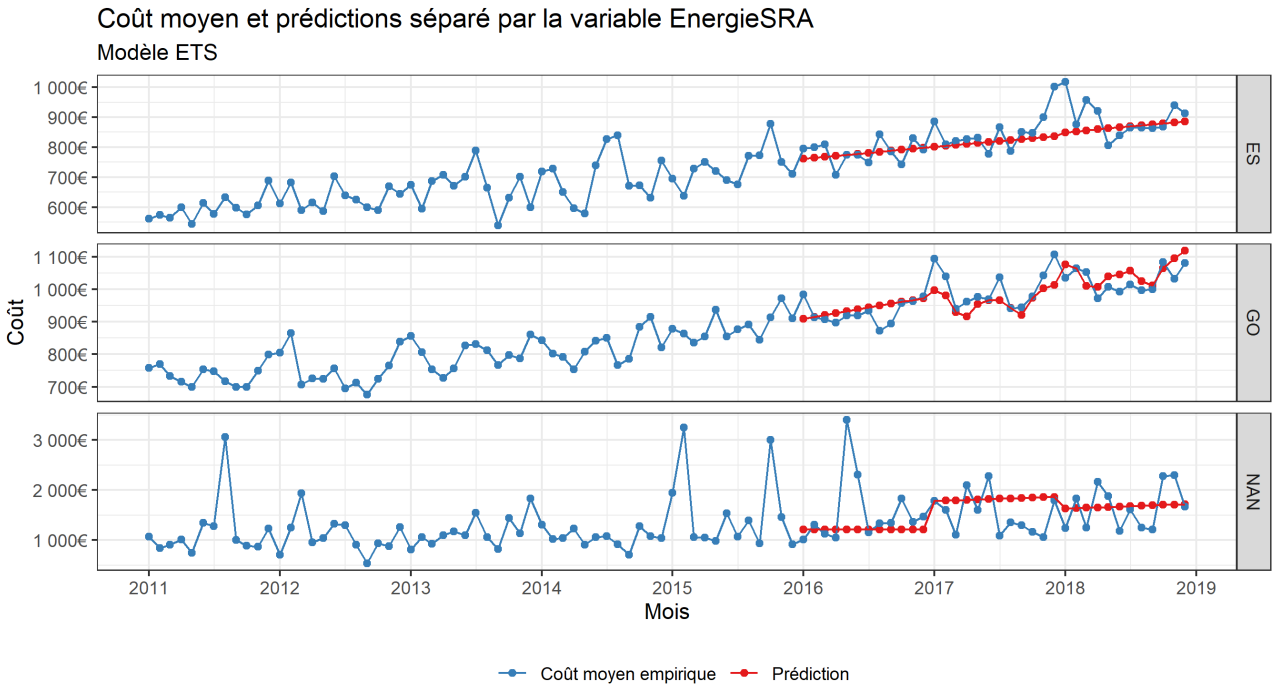
Ainsi, ces prédictions cadrent bien avec les données empiriques observées, sauf pour les outliers importants, ainsi que pour les données du milieu de l'année 2018. Ces erreurs de prédiction dans ces points précis sont communes à beaucoup de nos modèles. On peut ainsi considérer que ces montants étaient peu anticipables.
Regardons les résultats pour notre deuxième couple modèle-variable.

ETS-ENERGIE_SRABIS

La variable ENERGIE_SRA_BIS comporte trois modalités :

— ES pour les voitures essence.
— GO pour les voitures gasoil.
— NAN pour les autres

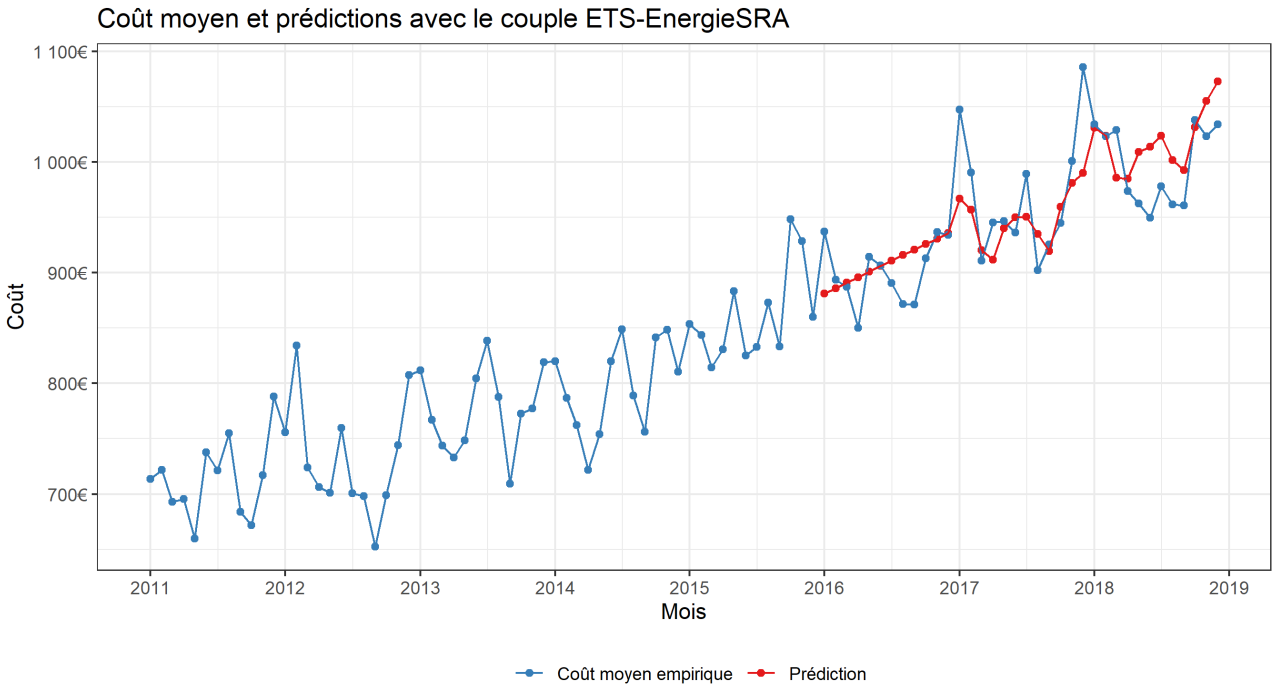
On obtient :



La modalité NAN est plutôt mal estimée, ce qui est dû aux outliers importants. La partie essence est modélisée avec une tendance linéaire simple, tandis que la partie gasoil a des erreurs de prédictions très faibles.
Nous avons la répartition des sinistres suivantes :

ENERGIE_SRABIS	Répartition du nombre de sinistres
ES	26.02%
GO	71.83%
NAN	2.15%

Ainsi, les erreurs relevées dans la partie NAN auront encore une fois très peu d'impacts dans les erreurs de prédictions finale.
Nous obtenons finalement la prédiction totale suivante :



5.7 Résultats des modèles dans la modélisation annuelle du coût moyen

Maintenant que l'on a obtenu nos prédictions mensuelles, nous voulons agréger ces prédictions pour construire nos prédictions annuelles.
En notant :

— $\hat{Y}_N$ la prédiction pour le coût moyen des pièces pour l'année N
— $\hat{y}_{i,N}$ la prédiction pour le coût moyen des pièces pour le mois i et l'année N.
— r_i la répartition empirique du nombre de sinistres pour le mois i.
On a : $r_i = \frac{\text{Nombre de sinistres pour le mois i}}{\text{Nombre de sinistres total}}$

On obtient ainsi : $\hat{Y}_N = \sum_{i=12}^{12} r_i \hat{y}_{i,N}$.
En particulier, on obtient, pour nos données, la répartition des sinistres suivantes :

Mois	**Répartition**
Janvier	8.05%
Février	7.17%
Mars	7.89%
Avril	8.12%
Mai	8.58%
Juin	9.72%
Juillet	9.04%
Août	7.61%
Septembre	8.57%
Octobre	8.68%
Novembre	8.31%
Décembre	8.25%

On obtient ainsi les prédictions annuelles suivantes

		Modèle TBATS		Modèle ETS	
Année	Coût moyen empirique	Prédiction	Erreur	Prédiction	Erreur
2011	712.85€				
2012	733.63€				
2013	777.79€				
2014	798.32€				
2015	863.13€				
2016	901.32€	899.34€	-0.22%	908.86€	+0.84%
2017	969.98€	938.26€	-3.27%	948.54€	-2.21%
2018	996.02€	1010.37€	+1.44%	1018.94€	+2.30%

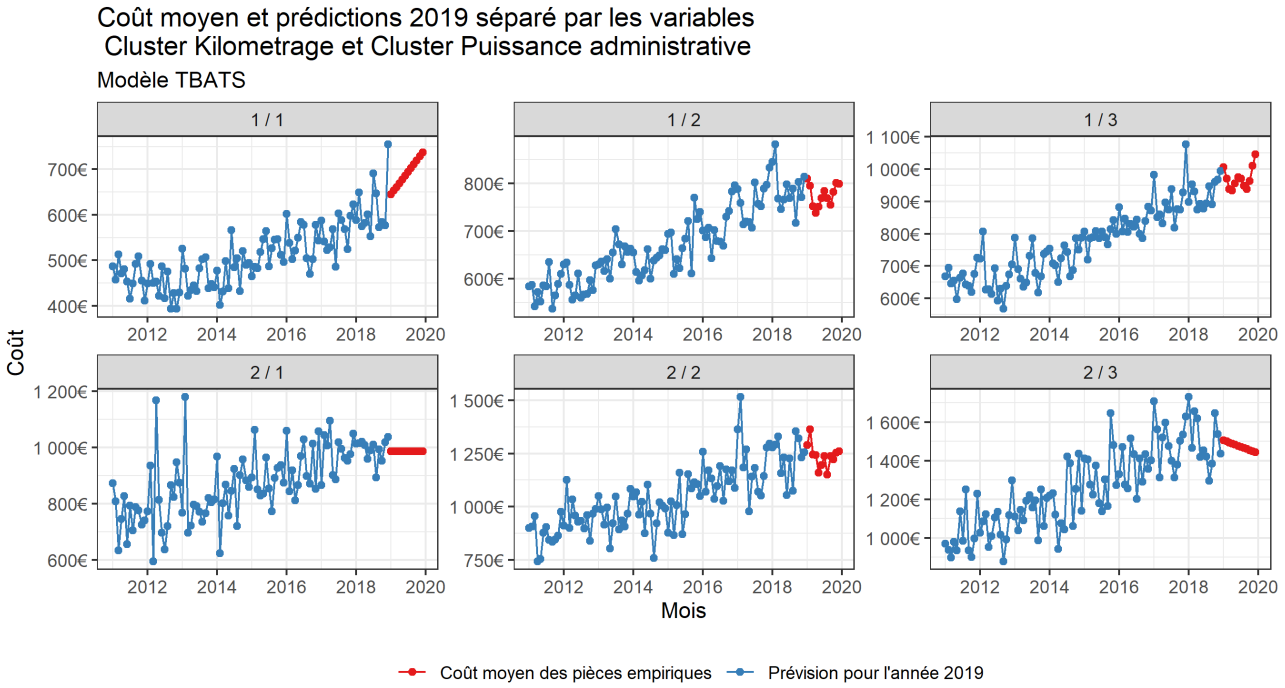
Ainsi, nous remarquons que les deux modèles prédisent précisément le coût moyen de l'année 2016, tandis que leurs performances sur les années 2017 et 2018 diffèrent. Cependant les erreurs pour l'année 2017 sont toutes deux négatives, et celles pour l'année 2018 sont toutes deux positives, ce qui peut être dû à nos données initiales.

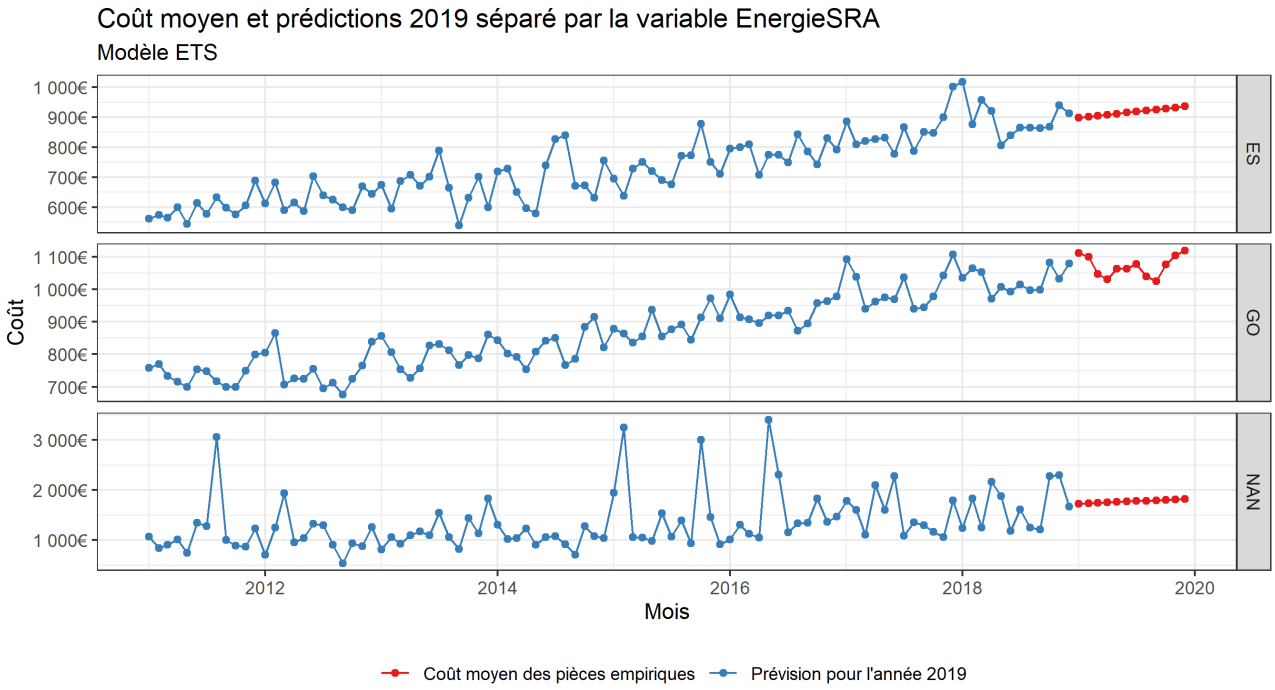
Après avoir jugé de la performance de nos deux meilleurs couples Modèle-variables nous allons déterminer les prédictions de ces couples pour l'année 2019.

5.8 Prédictions de l'année 2019

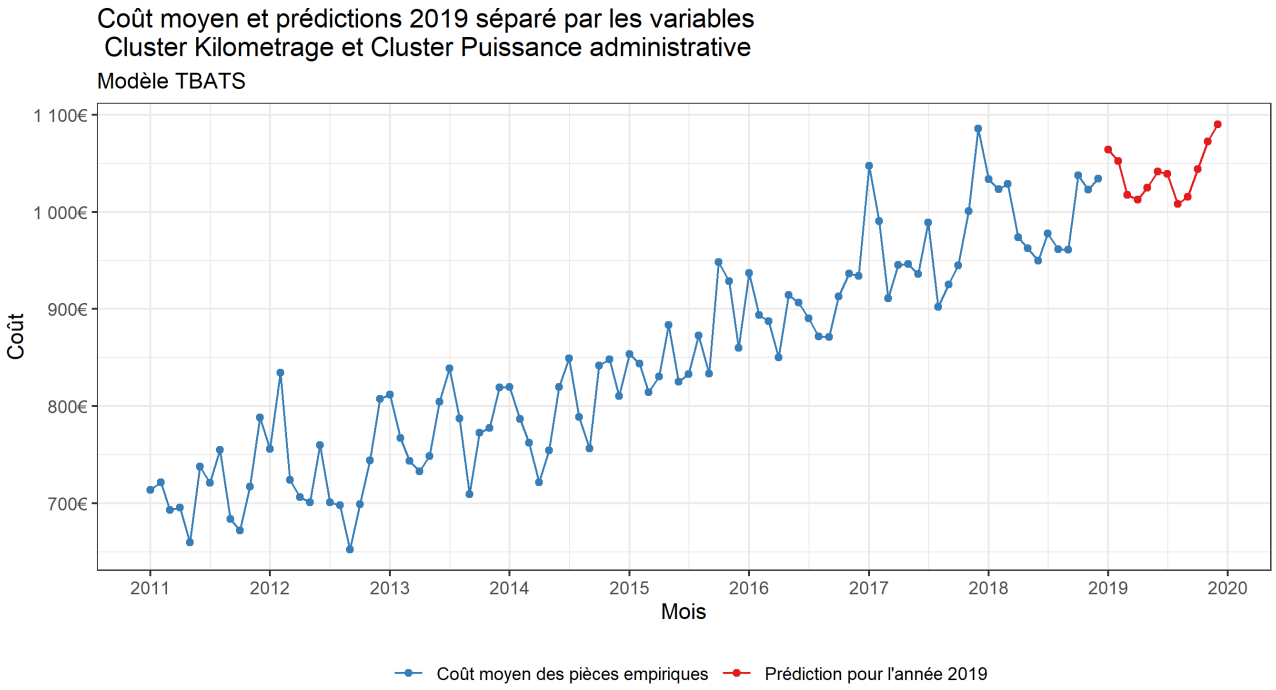
5.8.1 Prédictions mensuelles

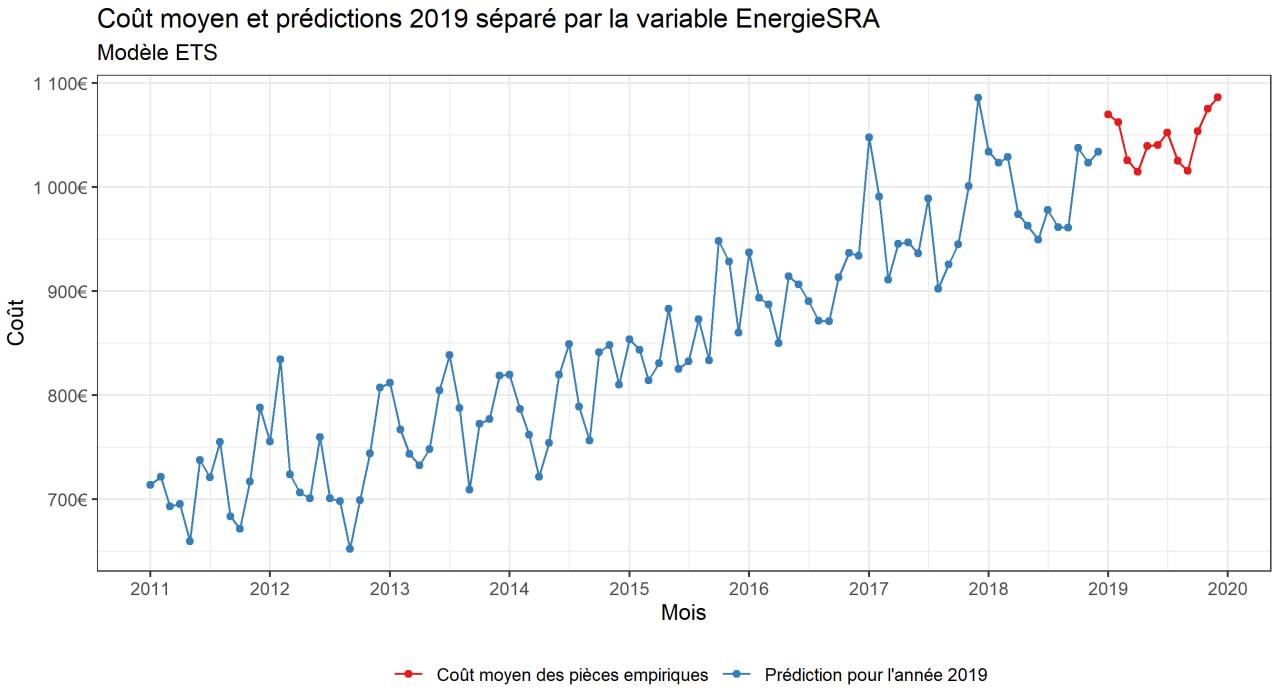
Nous avons les prédictions suivantes pour l'année 2019 :





On construit ensuite les prédictions mensuelles totales. On obtient les résultats ci dessous.





Nos deux prédictions présentent ainsi des similitudes, notamment dans la modélisation de la saisonnalité, où on retrouve les mêmes motifs pour les deux prédictions. Cependant, la prédiction avec le modèle TBATS semble plus faible que celle avec le modèle ETS.
Pour la prédictions annuelle 2019, on obtient les résultats ci-dessous.

5.8.2 Prédiction annuelle

Nous obtenons finalement les résultats suivants :

	Modèle TBATS		**Modèle ETS**	
Année	Coût moyen	Inflation	Coût moyen	Inflation
2011	712.85€		712.85€	
2012	733.63€	2.92%	733.63€	2.92%
2013	777.79€	6.02%	777.79€	6.02%
2014	798.32€	2.64%	798.32€	2.64%
2015	863.13€	8.12%	863.13€	8.12%
2016	901.32€	4.42%	901.32€	4.42%
2017	969.98€	7.62%	969.98€	7.62%
2018	996.02€	2.68%	996.02€	2.68%
2019	**1040.26€**	**4.44%**	**1046.62€**	**5.08%**

Nos deux prédictions sont ainsi assez proche,avec une différence de 0.64% entre les deux prédictions. Plus précisément,nous obtenons, avec deux modèles de séries temporelles différentes et deux séparations différentes, des prédictions assez proche. Nous pouvons donc avoir une bonne confiance en ces deux prédictions. Ce résultat est d'autant plus surprenant que l'inflation empirique était elle, très volatile, pouvant passer de 7.62% en 2017 à 2.68% en 2018.

Chapitre 6

Conclusion

Nous avons développé, au long de ce mémoire, une méthode pour estimer le coût moyen des pièces des sinistres auto-matériels. Cependant, les méthodes développées ici peuvent être utiles pour la prédiction de tous types de coûts moyens.
Pour cela, nous avons exposé en premier lieu deux méthodes permettant de diminuer le nombre de modalités d'une variable, afin d'obtenir des séries temporelles avec des tendances les plus prévisibles possible.
Nous avons ensuite détaillé 4 modèles mathématiques de modélisation de séries temporelle, des plus simples et classiques (modèles ARIMA et ETS), à des modèles plus complexes et plus récents (TBATS et MLP).

La méthode de prédiction séparant le coût moyen suivant les modalités d'une ou plusieurs variables s'est montrée utile pour produire des prédictions plus fines qu'avec la série temporelle totale. Cependant, il faut garder à l'esprit qu'en moyenne, les prédictions en séparant la base de données à l'aide d'une ou deux variables, fournissent des prédictions plus mauvaises que celles avec la série temporelle au global. Ce phénomène provient du fait que l'on a testé nos modèles avec beaucoup de variables ou regroupements de variables, sans que ceux ci soient forcement pertinents pour l'estimation du coût moyen des pièces.

L'utilisation de modèles complexes comme les modèles TBATS et MLP s'est montré intéressante pour plusieurs raisons.
Les modèles TBATS permettent une modélisation plus fine de la saisonnalité des séries temporelles, à l'aide des séries de Fourier notamment. Ce modèle produit les meilleurs estimations, même si les écarts avec les modèles ETS restent faible.
Les modèles MLP, utilisant des réseaux de neurones, ont montrés quant à eux des limites (prédictions comportant une composante aléatoire,temps de calculs long) et n'a pas produit de prédictions très précises. Cependant, les erreurs maximum se sont révélées plus faibles qu'avec les autres modèles.
Ces modèles restent donc intéressant à tester pour de futures estimations de coût moyen, si les capacités informatiques le permettent. En effet, dans notre cas nous avons utilisé seulement 200 réseaux de neurones par estimation, pour des contraintes de temps de calculs, tandis que 2000 ou même 20000 auraient fourni des résultats encore plus stables.

Notons qu'il existe des modèles, que nous n'avons pas développé dans ce mémoire, permettant, avec par exemple une base de données comportant deux variables explicatives A et B, de construire une estimation en moyennant les prédictions produites en séparant la base de données suivant les variables A et B, et sans séparer la base de données. Ces modèles se nomment souvent HTS (Hierarchical Time Series), et peuvent être un point intéressant à développer pour de futures études.

Annexe A

Bibliographie

Rob J Hyndman and George Athanasopoulos : *Forecasting: Principles and Practice.*

Nikolaos Kourentzesa, Devon K. Barrowa, Sven F. Cronea : *Neural network ensemble operators for time series forecasting*

Peter Ellis : *Better prediction intervals for time series forecasts*

Peter Ellis : *Why time series forecasts prediction intervals aren't as good as we'd hope*

Annexe B

Packages et fonctions R utilisés

Package forecast

— **auto.arima :** Prend en arguments une série temporelle, et ressort le meilleur modèle ARIMA.
— **ets :** Prend en arguments une série temporelle, et ressort le meilleur modèle ETS.
— **tbats :** Prend en arguments une série temporelle, et ressort le meilleur modèle TBATS.
— **hts :** Prend en arguments une série temporelle, et ressort le meilleur modèle HTS (non développé dans ce mémoire).
— **forecast :** Prend en argument un modèle du package forecast, et ressors le nombre de prédictions souhaité.

Package nnfor

— **mlp :** Prend en argument une série temporelle et ressors le meilleur modèle MLP. Dans ce mémoire, nous avons paramétré manuellement beaucoup de paramètre comme le nombre de neurones dans la couche cachée, le nombre de réseaux entraînés, ainsi que la méthode d'agrégation.
— **forecast :** Prend en argument un modèle MLP et ressors le nombre de prédictions souhaité.

Package de base :

— **kmeans** : Prend en argument une matrice numérique et un nombre de clusters, et ressors un objet de la classe kmeans comprenant notamment la liste des clusters ligne à ligne

dplyr

Package utilisé pour le traitement des données.

ggplot2

Package utilisé pour produire les graphiques