Main street in downtown Ames, Photo by Tim Kiser

House Prices - Advanced Regression Techniques

Predict sales prices and practice feature engineering

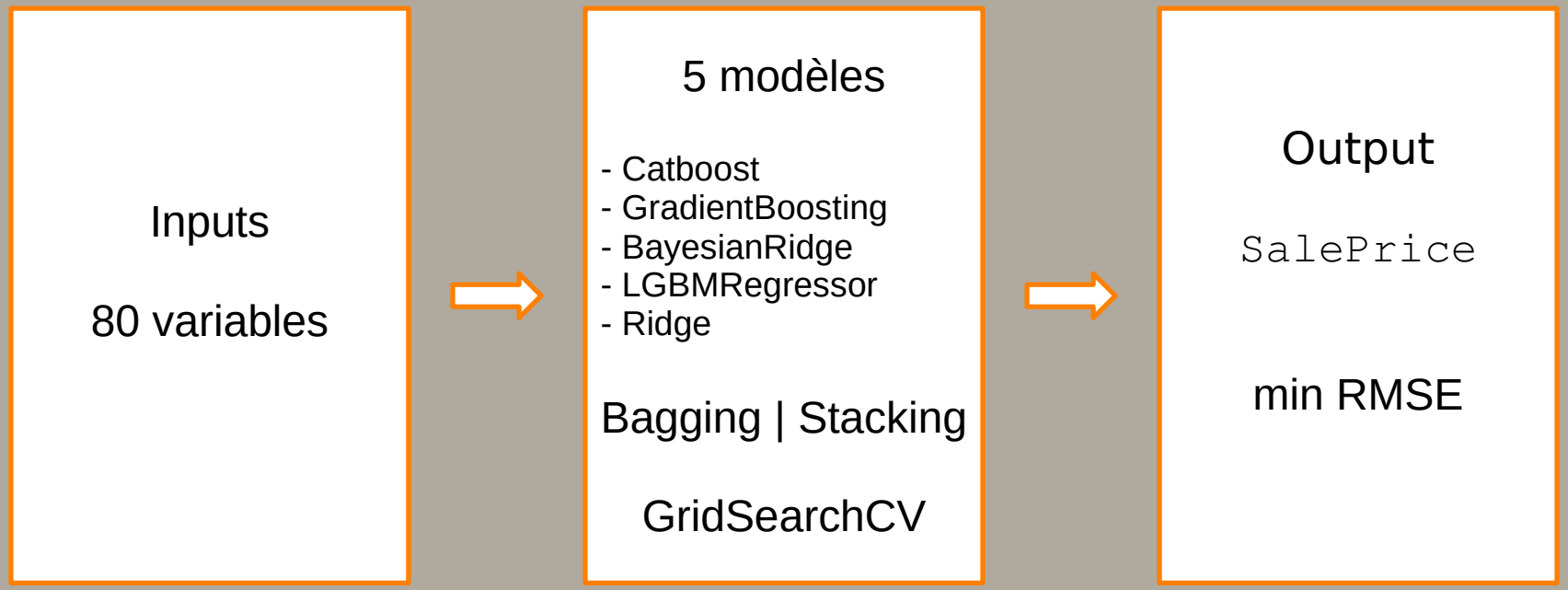
COMPETENCES EVALUÉES

- Rédiger une note méthodologique afin de communiquer sa démarche de modélisation
- Utiliser un logiciel de version de code pour assurer l’intégration du modèle
- Présenter son code aux standards PEP 8
- Enrichir les réalisations d'autres membres de la communauté de professionnels

PROBLÉMATIQUE

Notre objectif est d'utiliser l'ensemble des données sur les logements d'Ames, Iowa pour construire un modèle d'apprentissage automatique capable de prédire le prix de vente d'une propriété résidentielle.

PROBLÉMATIQUE



MÉTHODOLOGIE

-> Erreurs de formatage

-> Valeurs manquantes

-> Erreurs de typage

-> Exploratory data analysis (EDA)

-> Outliers

-> Feature engineering

-> Modélisation

-> Tuning

-> Soumission des résultats

DONNÉES

80 variables au total – 1460 observations de 2006 à 2010:

- 46 variables catégorielles allant de 2 à 28 classes
    - 23 nominales: sous-classes de bâtiment
    - 23 ordinales: mesure se qualité

- 14 discrètes: nombre d'éléments présents dans la maison
    - cuisines, salles de bains etc.
    - année ou mois

- 20 continues:
    - la taille du terrain
    - la superficie totale du logement en pieds carrés

L'ensemble de données sur le logements d'Ames a été collecté par le Dr Dean De Cock, professeur de statistiques à la Truman State University, en 2011.

DATA PROCESSING

Erreurs de formatage

```
BsmtFinType1: Rating of basement finished
area

       GLQ     Good Living Quarters
       ALQ     Average Living Quarters
       BLQ     Below Average Living Quarters
       Rec     Average Rec Room
       LwQ     Low Quality
       Unf     Unfinshed
       NA No Basement
```

Valeurs manquantes

- variables numériques : `KNNImputer()`
- variables catégoriques : `mode()`

Erreurs de typage

- `Float` to `int64`: `BsmtFullBath`, `GarageYrBlt`, `FullBath` etc.

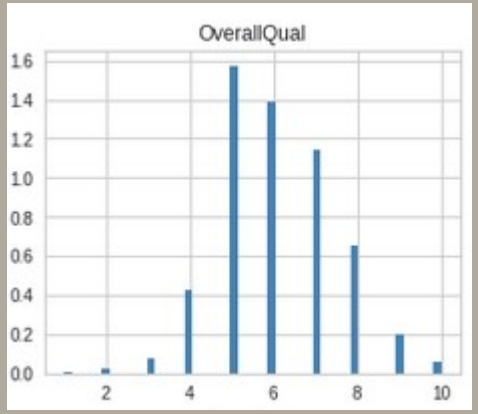
- `int64` to `object` : `MSSubClass`, `OverallQual`, `OverallCond`

- `object` to `int64`: variables ordinales

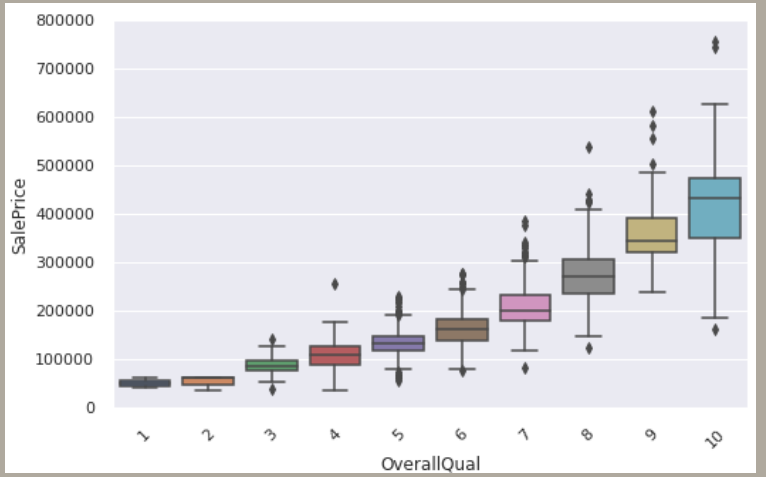
```
ExterQual ['Gd' 'TA' 'Ex' 'Fa']
ExterCond ['TA' 'Gd' 'Fa' 'Po' 'Ex']
BsmtQual ['Gd' 'TA' 'Ex' 'None' 'Fa']
BsmtCond ['TA' 'Gd' 'None' 'Fa' 'Po']
BsmtExposure ['No' 'Gd' 'Mn' 'Av'
'None']
HeatingQC ['Ex' 'Gd' 'TA' 'Fa' 'Po']
KitchenQual ['Gd' 'TA' 'Ex' 'Fa']
FireplaceQu ['None' 'TA' 'Gd' 'Fa' 'Ex'
'Po']
GarageQual ['TA' 'Fa' 'Gd' 'None' 'Ex'
'Po']
GarageCond ['TA' 'Fa' 'None' 'Gd' 'Po'
'Ex']
PoolQC ['None' 'Ex' 'Fa' 'Gd']
```

DATA EXPLORATION

Distribution des variables numériques

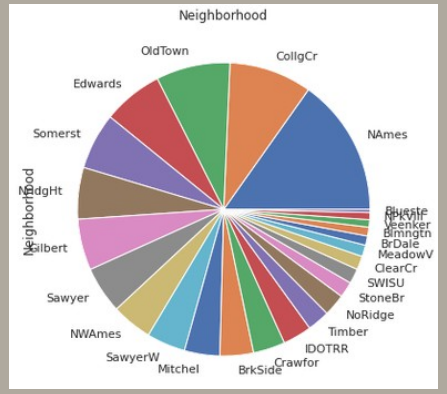


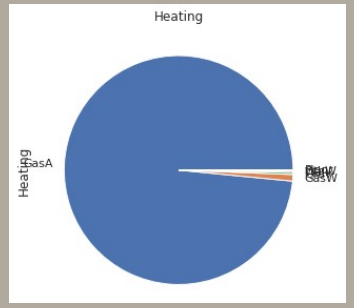
Distribution des variables numériques



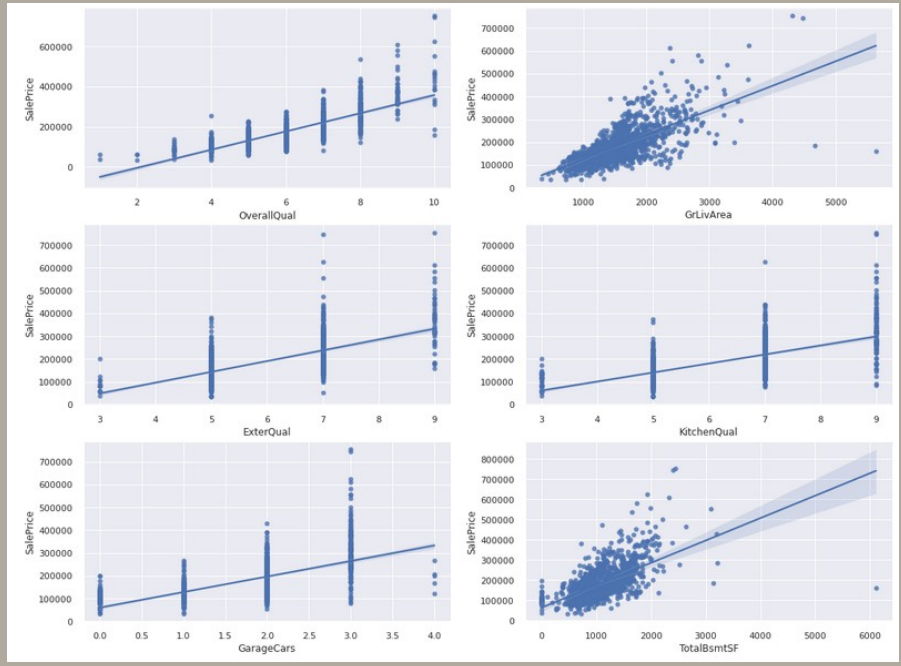
DATA EXPLORATION

Distribution des variables catégoriques

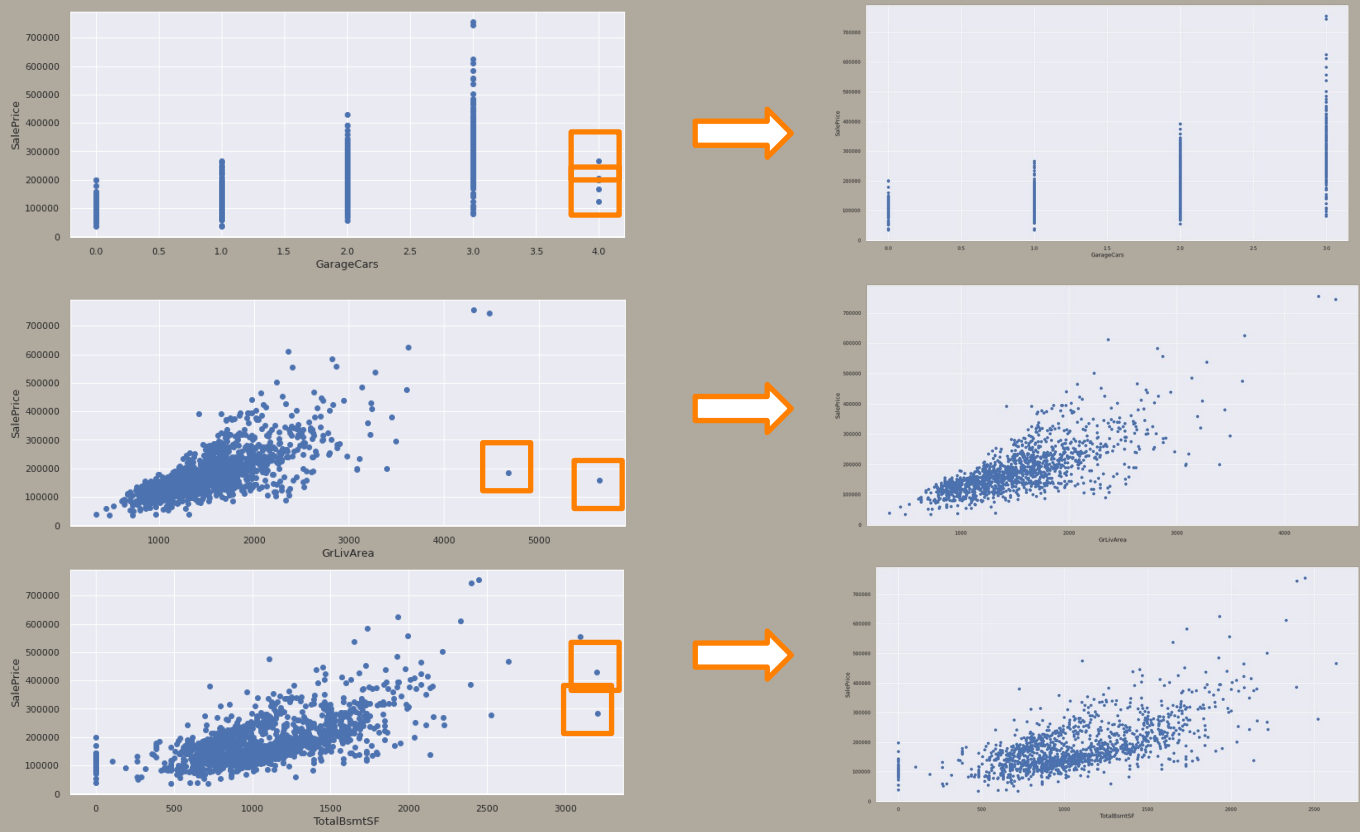




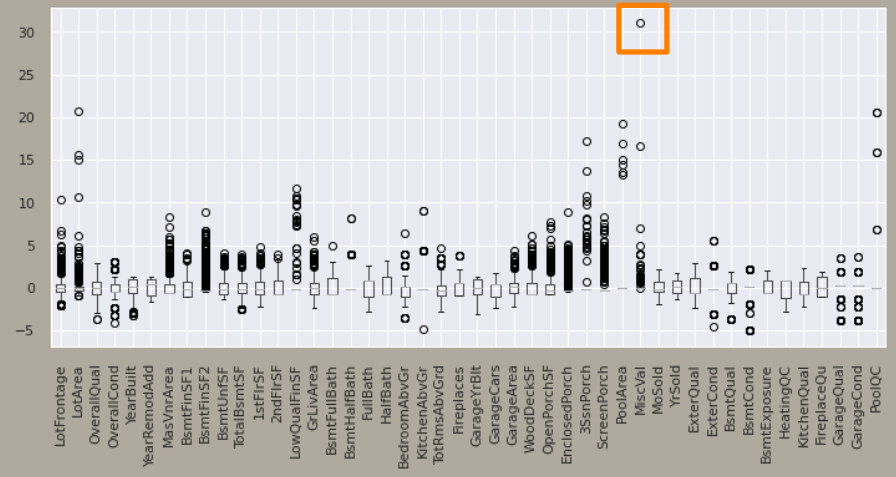
Corrélation des variables numériques à `SalePrice`

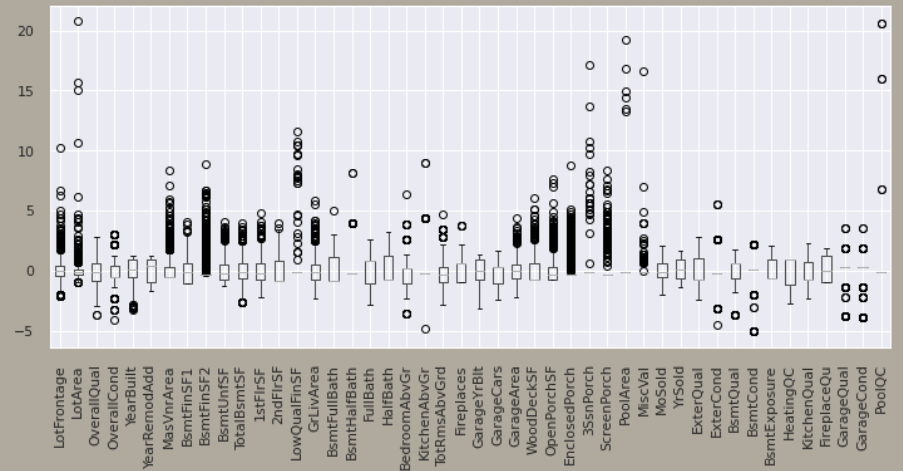


OUTLIERS



OUTLIERS


30
25
20
15
10
5
0
−5
LotFrontage
LotArea
OverallQual
OverallCond
YearBuilt
YearRemodAdd
MasVnrArea
BsmtFinSF1
BsmtFinSF2
BsmtUnfSF
TotalBsmtSF
1stFlrSF
2ndFlrSF
LowQualFinSF
GrLivArea
BsmtFullBath
BsmtHalfBath
FullBath
HalfBath
BedroomAbvGr
KitchenAbvGr
TotRmsAbvGrd
Fireplaces
GarageYrBlt
GarageCars
GarageArea
WoodDeckSF
OpenPorchSF
EnclosedPorch
3SsnPorch
ScreenPorch
PoolArea
MiscVal
MoSold
YrSold
ExterQual
ExterCond
BsmtQual
BsmtCond
BsmtExposure
HeatingQC
KitchenQual
FireplaceQu
GarageQual
GarageCond
PoolQC

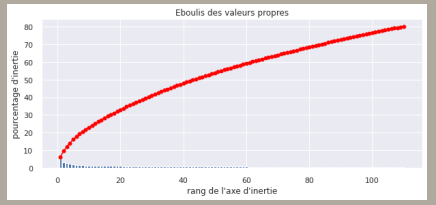

20
15
10
5
0
−5
LotFrontage
LotArea
OverallQual
OverallCond
YearBuilt
YearRemodAdd
MasVnrArea
BsmtFinSF1
BsmtFinSF2
BsmtUnfSF
TotalBsmtSF
1stFlrSF
2ndFlrSF
LowQualFinSF
GrLivArea
BsmtFullBath
BsmtHalfBath
FullBath
HalfBath
BedroomAbvGr
KitchenAbvGr
TotRmsAbvGrd
Fireplaces
GarageYrBlt
GarageCars
GarageArea
WoodDeckSF
OpenPorchSF
EnclosedPorch
3SsnPorch
ScreenPorch
PoolArea
MiscVal
MoSold
YrSold
ExterQual
ExterCond
BsmtQual
BsmtCond
BsmtExposure
HeatingQC
KitchenQual
FireplaceQu
GarageQual
GarageCond
PoolQC

FEATURES ENGINEERING

+ Features

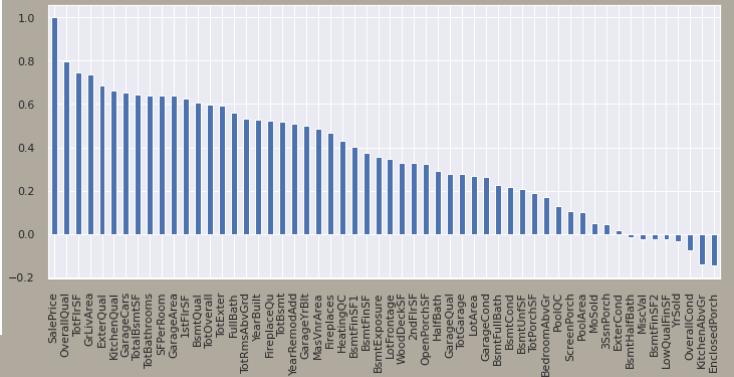
SFPerRoom
TotOverall
TotGarage
TotExter
TotBsmt
TotBathrooms
BsmtFinType
BsmtFinSF
TotFlrSF
TotPorchSF

+



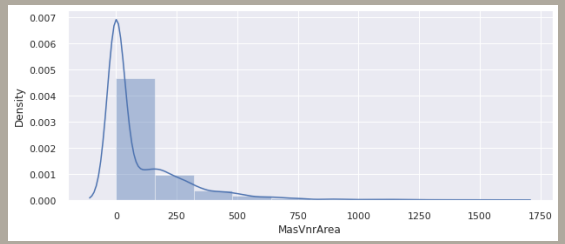
- Features

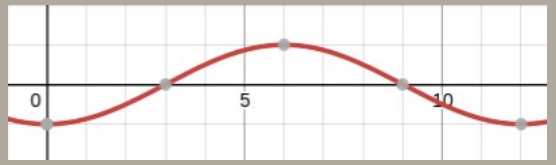
	level_0	level_1	corr_coeff
28	TotFlrSF	GrLivArea	1.00
26	GarageQual	TotGarage	0.99
24	GarageCond	TotGarage	0.99
22	GarageQual	GarageCond	0.95
20	BsmtQual	TotBsmt	0.94
18	BsmtFinSF	BsmtFinSF1	0.93
16	PoolQC	PoolArea	0.91
14	GarageArea	GarageCars	0.89
12	Fireplaces	FireplaceQu	0.86
10	ExterQual	TotExter	0.86
8	TotRmsAbvGrd	GrLivArea	0.83
6	TotRmsAbvGrd	TotFlrSF	0.82
4	TotBsmt	BsmtCond	0.82
2	YearBuilt	GarageYrBlt	0.80
0	TotalBsmtSF	1stFlrSF	0.80



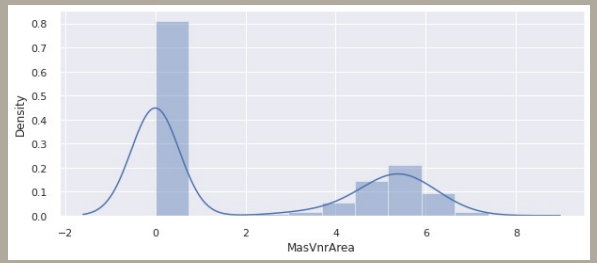
FEATURES ENGINEERING

Skewness: `log(x+1)`





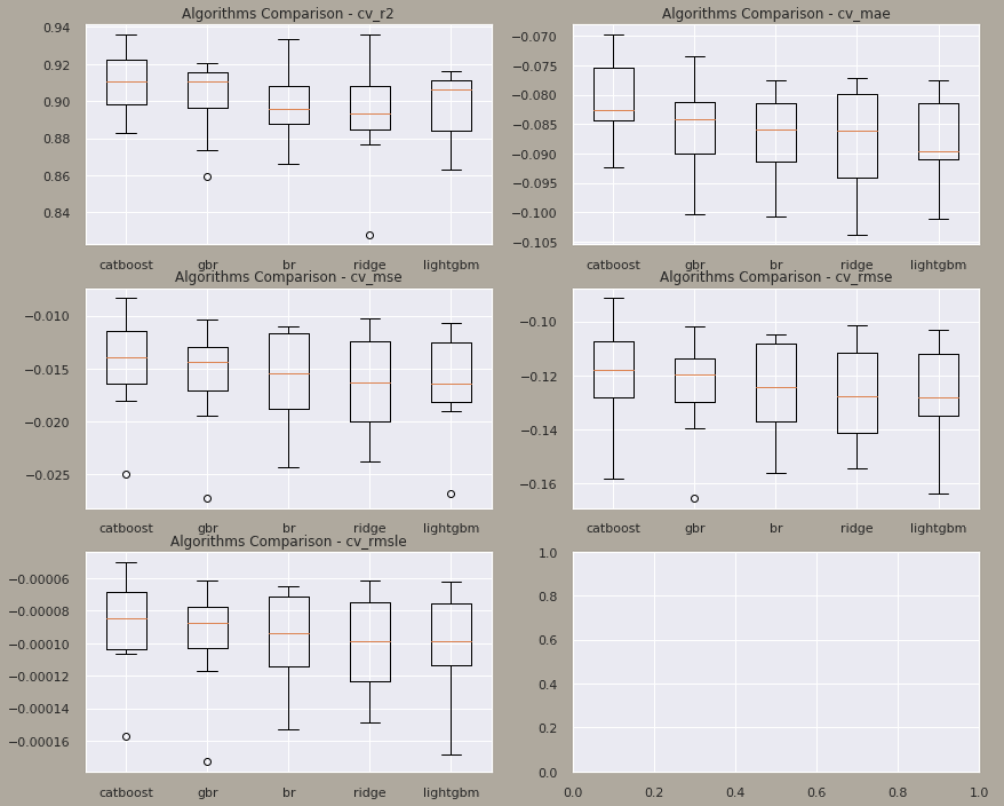
`- Cos(0.5236  x)`



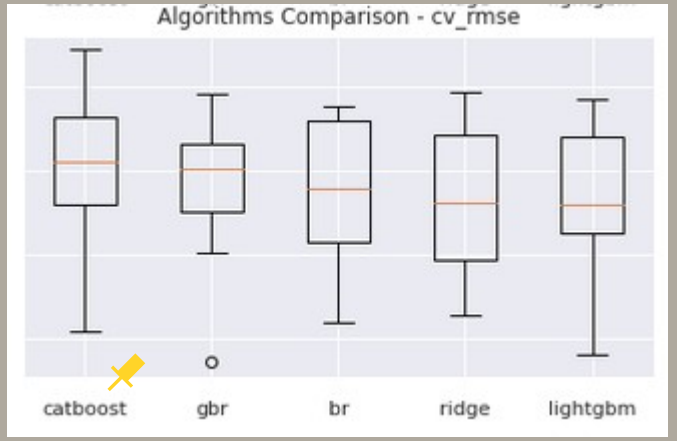
MODÉLISATION

- Encoding ✅
- Normalisation ✅
- `log(y)` ✅
- Split train | test ✅

Classification des modèles

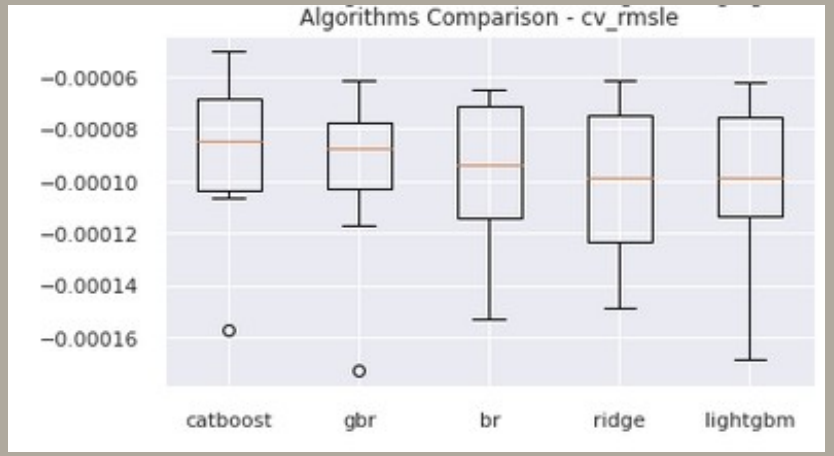


MODÉLISATION




```
catboost cv_rmsle: mean =  0.000088
std = 0.000030
catboost cv_rmse: mean =  0.118712
std = 0.018860
catboost cv_mae : mean =  0.080981
std = 0.007297
catboost cv_mse : mean =  0.014448
std = 0.004666
catboost cv_r2  : mean =  0.909867
std = 0.017926
catboost cv_rmse: exp(average
error) =  1.126045
```

Baseline:
catboost
regressor



MODÉLISATION

Baseline: catboost regressor

```
catboost cv_rmsle: mean =  0.000088 std = 0.000030
catboost cv_rmse: mean =  0.118712 std = 0.018860
catboost cv_mae : mean =  0.080981 std = 0.007297
catboost cv_mse : mean =  0.014448 std = 0.004666
catboost cv_r2  : mean =  0.909867 std = 0.017926
catboost cv_rmse: average error =  1.126045
```

	Id	SalePrice
0	1461	122280.80
1	1462	156660.81
2	1463	187629.83
3	1464	192667.31
4	1465	178686.50
...	...	...
1454	2915	84055.80
1455	2916	82374.96
1456	2917	154157.58
1457	2918	122527.66
1458	2919	208587.87

MODÉLISATION

Bagging | Stacking

```
models = {
    "catboost": CatBoostRegressor(),
    "gbr": GradientBoostingRegressor(),
    "br": BayesianRidge(),
    "lightgbm": LGBMRegressor(),
    "ridge": Ridge()
}
```

Cross validation score
cv = 10

```
----------
catboost
mean error: 1.1255705021722968
std error : 0.019166088729885393
----------
gbr
mean error: 1.1313293560334636
std error : 0.018247295842752285
----------
br
mean error: 1.1309322682595433
std error : 0.01677582200834476
----------
lightgbm
mean error: 1.1340964763343142
std error : 0.01605738440024673
----------
ridge
mean error: 1.1340701486514333
std error : 0.01579574106419592
```

Metrics = -exp (RMSE)

```
final_predictions = (
    0.4 * y_pred +
    0.2 * pred_gbr +
    0.2 * pred_br +
    0.1 * pred_ridge +
    0.1 * pred_lightgbm)
```

	Id	SalePrice
0	1461	122045.73
1	1462	158389.52
2	1463	184292.28
3	1464	194699.75
4	1465	185866.32
...	...	...
1454	2915	84402.42
1455	2916	79346.58
1456	2917	156052.84
1457	2918	122624.40
1458	2919	208398.34

MODÉLISATION

GridSearchCV
cv = 5

- hyper paramètres

- predict() * 5

Cross validation
score
cv = 10

```
----------
catboost
mean error: 1.1240598432253028
std error : 0.01738574116687094
----------
gbr
mean error: 1.127137357413304
std error : 0.01594005333252564
----------
br
mean error: 1.1309322170966687
std error : 0.01677577289463734
----------
lightgbm
mean error: 1.128179739246058
std error : 0.016585223136415268
----------
ridge
mean error: 1.1334127886973446
std error : 0.01584938403087582
```

Metrics = -exp (RMSE)

```
final_predictions = (
    0.4 * y_pred +
    0.2 * pred_gbr +
    0.2 * pred_br +
    0.1 * pred_ridge +
    0.1 * pred_lightgbm)
```

	Id	SalePrice
0	1461	122140.47
1	1462	161014.74
2	1463	184077.83
3	1464	194726.37
4	1465	188387.42
...	...	...
1454	2915	84740.62
1455	2916	79044.91
1456	2917	160046.84
1457	2918	121327.59
1458	2919	208635.44

CONCLUSION

Steps	Score	Rank %	Rank
Baseline catboost	0.12301	7.11%	442
Bagging Ensemble	0.12386	7.17%	445
Add Features	0.12213	7.04%	439
Drop Features	0.12235	7.06%	440
Baseline : Add Features + Features Elimination + Outliers	0.12562	7.30%	451
Bagging : Add Features + Features Elimination + Outliers	0.12296	7.11%	442
Baseline with PCA	0.1294	7.61%	465
Bagging with PCA	0.12463	7.24%	448
Bagging with PCA and GridSearchCV	0.12364	7.15%	444
Best score:	0.12213		

- AutoML avec pycaret
- Ajout de colonnes
- Bagging | Stacking
- GridSearchCV

- supprimer les variables

https://www.kaggle.com/catherine83/kernel642227a72

Thanks !
Any
questions ?