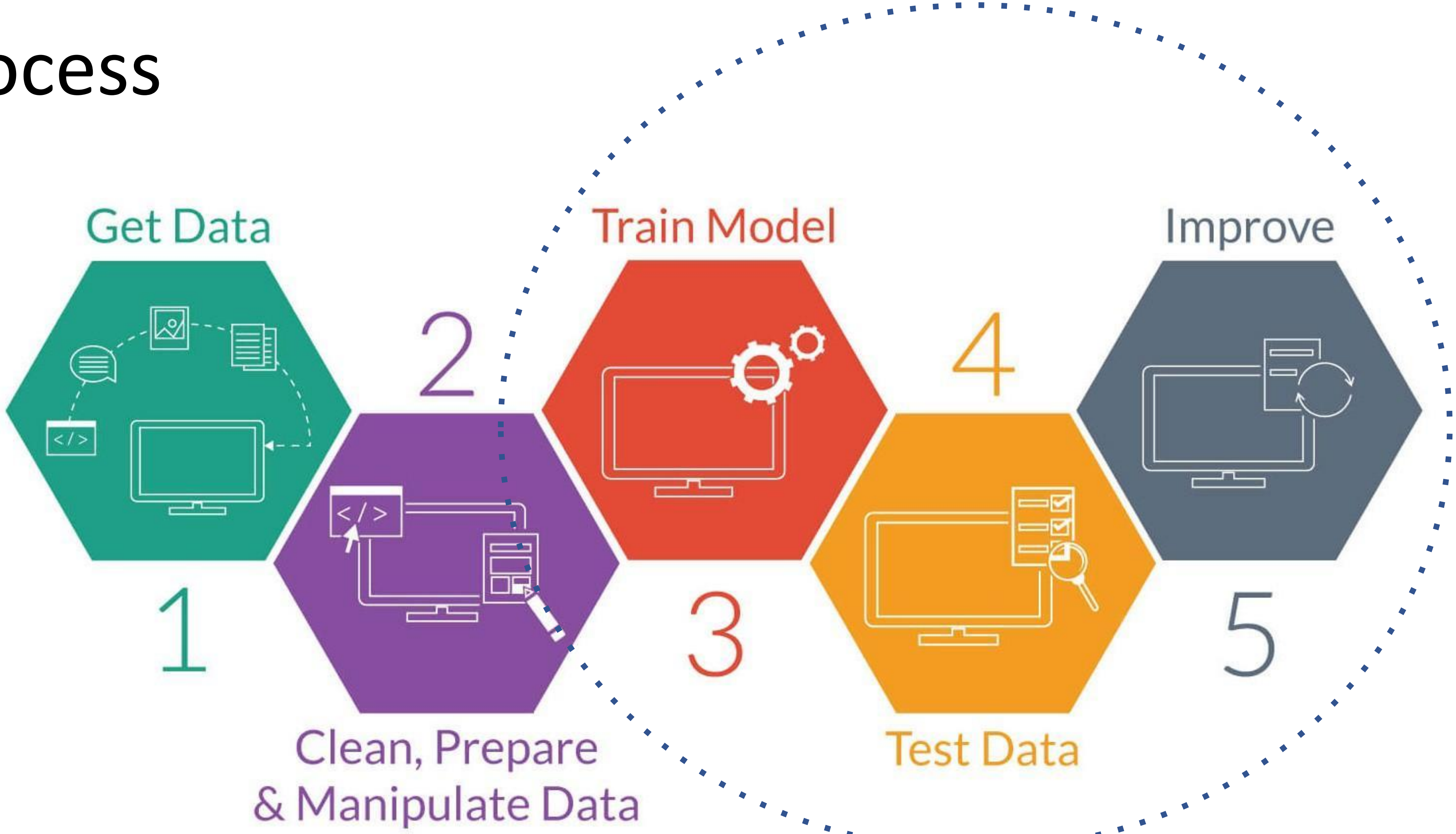
Modelization

Process



Get Data
1
2
Clean, Prepare
& Manipulate Data
Train Model
3
4
Test Data
Improve
5

Process

1. Fitting of a « Full-Model » on the Training Dataset
2. Optimization of the Model (Automatic and Manual)
3. Validation on the Testing Dataset
4. Benchmarking with other models

Theory

Model

Logistic Regression is used to predict the probability of an event given a set of explainary variables (X1,…,Xp) with the equation bellow :

$$\mathrm{logit}(P(y=1)) = log\left(\frac{P(y=1)}{1-P(y=1)}\right) = log\left(\frac{P(y=1)}{P(y=0)}\right) = \theta_0 + \theta_1 x_1 + \ldots + \theta_p x_p$$

where *P(y=1)* indicates the probability of an event (e.g., churn), Op are the regression coefficients associated with the reference group and the x_i explanatory variables.
Also, and Oo represent the reference group constituted by those individuals presenting the reference level of each and every variable x1...p.

Probabilities

- Probability is the ratio between the number of events favorable to some outcome and the total number of events, constrained between 0 and 1.

$$probability = \frac{chance}{1 + chance}$$

- It's extracted from the Logistic Regression equation as :

$$P\left(y^{(i)} = 1 \mid x^{(i)}, \theta\right) = \frac{1}{1 + exp(-(\theta_0 + \theta_1 x_1^{(i)} + \ldots + \theta_p x_p^{(i)}))}$$

Odds Ratio

- Odds are the ratio between probabilities: the probability of an event favorable to an outcome and the probability of an event against the same outcome, constrained between 0 and infinity.

$$odds = \frac{p}{(1-p)}$$

- And odds ratio is the ratio between odds. Therefore, a large odds ratio (OR) can represent a small probability and vice-versa.

$$\frac{odds_{x_j+1}}{odds} = \frac{exp\left(\theta_0 + \theta_1 x_1 + \ldots + \theta_j(x_j+1) + \ldots + \theta_p x_p\right)}{exp\left(\theta_0 + \theta_1 x_1 + \ldots + \theta_j x_j + \ldots + \theta_p x_p\right)} = exp\left(\theta_j(x_j+1) - \theta_j x_j\right) = exp\left(\theta_j\right)$$

- The result is the impact of each variable on the odds ratio of the observed event of interest. The main advantage is to avoid confounding effects by analyzing the association of all variables together.

Scorecard

- The idea is to give to each modality their own score, calculated with the following expression :

$$\text{Note} = \frac{\text{Coef}_{Modalité} - \text{Coef\_Min}_{Variable}}{\sum_{Chaque\ variable}(\text{Coef\_Max}_{Variable} - \text{Coef\_Min}_{Variable})} * 1000$$

Model Evaluation

- Fitting Quality : Accuracy, Likelihood, Wald, AIC,R^2…
- Prediction Quality : Lift, Gini, Roc, AUC…
- Business Criteria

Features Selection

Automatic

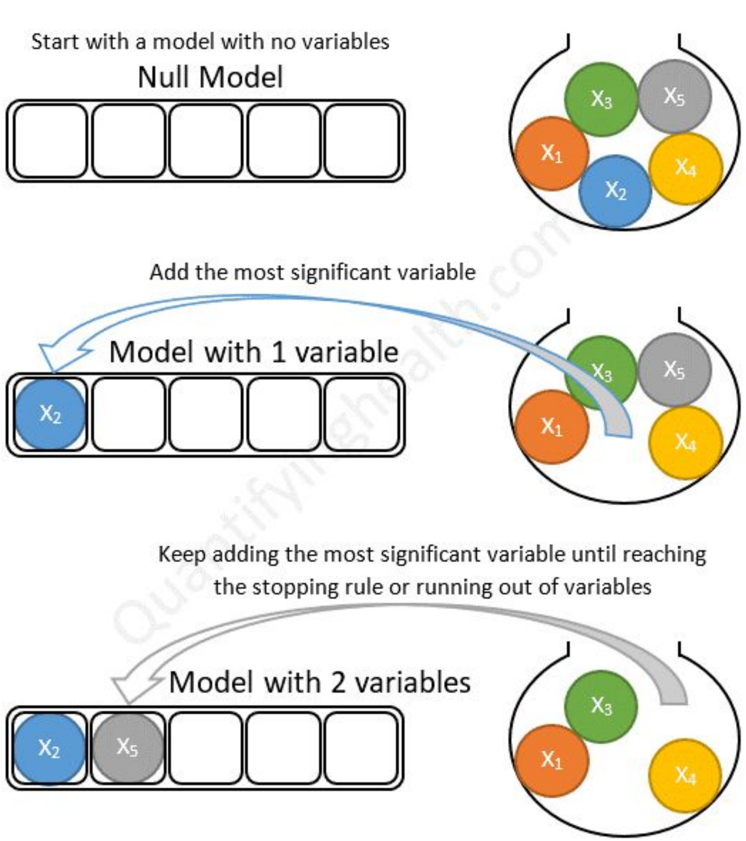
- Backward
- Forward

Manual

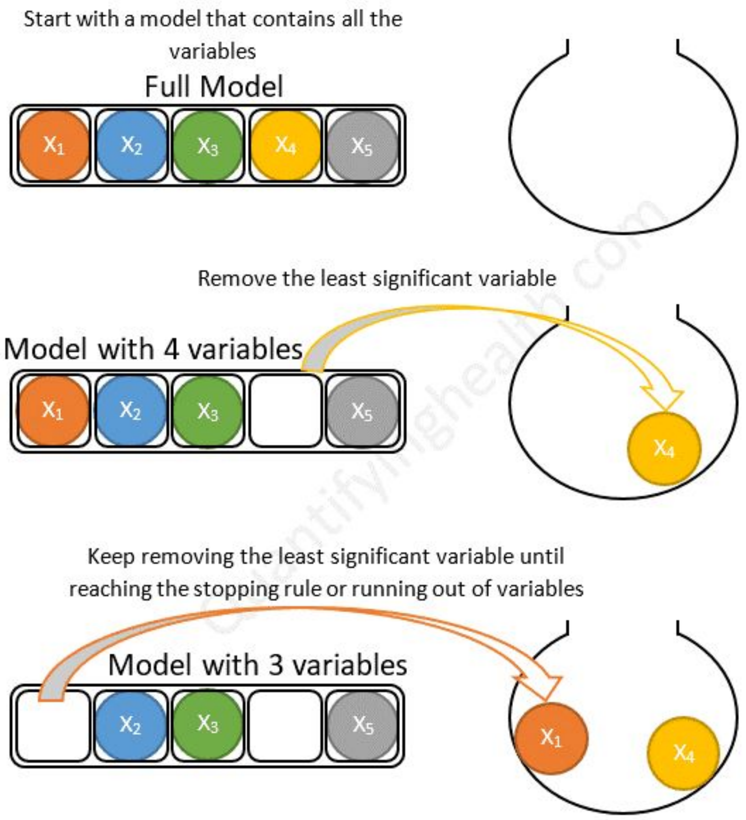
- Bivariate and multivariate Analysis
- Business knowledge

Features Selection- Stepwise

Forward stepwise selection example with 5 variables:



Backward stepwise selection example with 5 variables:



Other possible ML models

- Every models that gives classification : Decision Tree, SVM, Random Forest, XGBoost, etc
- Interpretability can be a decisive factor.

Example

Example

- Dataset : https://www.kaggle.com/competitions/customer-churn-prediction-2020/overview
- Telecom compagny wants to predict churn of their clients, in order to organize their email campaign and better understand the reasons behind it.
- Data : seniority, residence, time and area of calls, number of voicemails, type of plans they subscribe to.

Example – first fit

```python
from sklearn.linear_model import LogisticRegression

model_lr = LogisticRegression(solver='liblinear')
model_lr.fit(X_train_norm, y_train)

log_odds = model_lr.coef_[0]
pd.DataFrame(log_odds,
             X_train_norm.columns,
             columns=['coef'])\
            .sort_values(by='coef', ascending=False)
```

	coef
international_plan	2.136796
total_night_minutes	0.514232
total_night_charge	0.512446
number_customer_service_calls	0.509975
total_intl_charge	0.174766
total_intl_minutes	0.031106
total_day_minutes	0.016716
number_vmail_messages	0.010136
total_eve_minutes	0.004505
total_eve_charge	0.000849
account_length	-0.001064
total_day_calls	-0.005485
total_day_charge	-0.026653
total_intl_calls	-0.104053
total_night_calls	-0.892375
total_eve_calls	-1.234527
voice_mail_plan	-1.648523

Example – Interpretation

```
odds = np.exp(model_lr.coef_[0])
pd.DataFrame(odds,
             X_train_norm.columns,
             columns=['coef'])\
            .sort_values(by='coef’, ascending=False)
```

	coef
international_plan	8.472248
total_night_minutes	1.672354
total_night_charge	1.669369
number_customer_service_calls	1.665250
total_intl_charge	1.190968
total_intl_minutes	1.031595
total_day_minutes	1.016857
number_vmail_messages	1.010188
total_eve_minutes	1.004516
total_eve_charge	1.000849
account_length	0.998937
total_day_calls	0.994530
total_day_charge	0.973700
total_intl_calls	0.901177
total_night_calls	0.409682
total_eve_calls	0.290972
voice_mail_plan	0.192334

For every client that subscribed to the international plan, the odds for churning are 8,5 times as large as the odds for not churning when all other variables are held constant.

As variable “total_night_charge” increases by one unit, the odds for churning are over 1,7 as large as the odds for not churning.

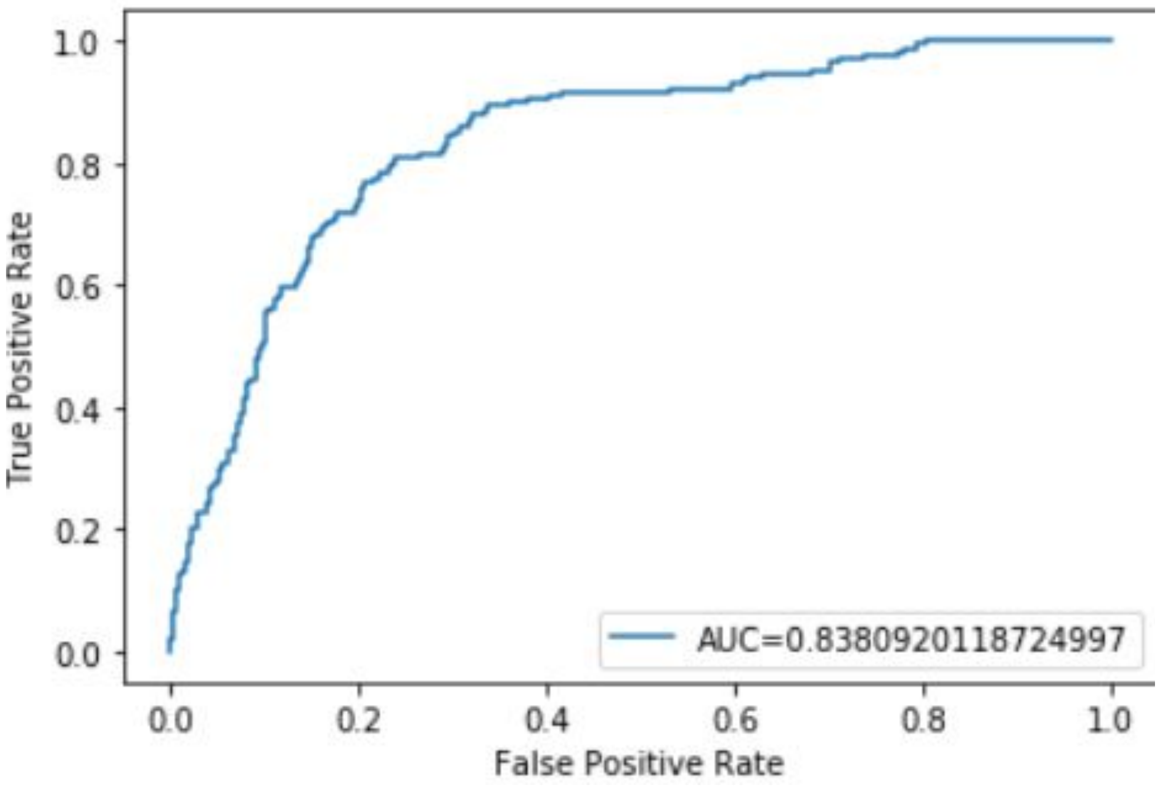
Model Evaluation – Classification

```python
from sklearn.metrics import confusion_matrix,
classification_report, accuracy_score

print('Accuracy: ')
print('{}'.format(accuracy_score(y_val, y_pred)))
print('Classification report: ')
print('{}'.format(classification_report(y_val, y_pred)))
print('Confusion Matrix')
print('{}'.format(confusion_matrix(y_val, y_pred)))
```

```
Accuracy:
0.8737254901960785
Classification report:
              precision    recall  f1-score   support

           0       0.89      0.98      0.93      1107
           1       0.56      0.18      0.28       168

    accuracy                           0.87      1275
   macro avg       0.73      0.58      0.60      1275
weighted avg       0.85      0.87      0.84      1275

Confusion Matrix
[[1083   24]
 [ 137   31]]
```

Model Evaluation – ROC Curve & AUC

```python
from sklearn.metrics import roc_curve,
roc_auc_score

y_pred_proba =
model_lr.predict_proba(X_val_norm)[::,1]

fpr, tpr, _ = roc_curve(y_val,  y_pred_proba)

auc = roc_auc_score(y_val, y_pred_proba)

plt.plot(fpr,tpr,label="AUC="+str(auc))
plt.ylabel('True Positive Rate')
plt.xlabel('False Positive Rate')
plt.legend(loc=4)
plt.show()
```



Features Selection - Colinearity

	account_length	international_plan	voice_mail_plan	number_vmail_messages	total_day_minutes	total_day_calls	total_day_charge	total_eve_minutes	number_customer_service_calls	total_intl_calls	total_intl_charge	total_eve_calls	total_eve_charge	total_night_minutes	total_night_calls	total_night_charge	total_intl_minutes
account_length	1.000000	0.040827	0.002778	-0.005711	-0.022742	0.016283	-0.022735	-0.025878	0.012635	0.028990	0.010828	0.005329	-0.025873	-0.021610	-0.006445	-0.021657	0.010933
international_plan	0.040827	1.000000	0.002776	-0.000133	0.018833	-0.001316	0.018839	0.004479	-0.019765	-0.001989	0.018457	-0.012504	0.004482	-0.007317	0.013870	-0.007323	0.018446
voice_mail_plan	0.002778	0.002776	1.000000	0.953880	-0.012776	-0.002650	-0.012783	0.020704	-0.019309	-0.012405	0.010257	-0.007505	0.020709	0.003341	0.005222	0.003361	0.010206
number_vmail_messages	-0.005711	-0.000133	0.953880	1.000000	-0.014438	0.000085	-0.014445	0.018572	-0.009096	0.000647	0.009337	-0.004580	0.018583	0.011243	0.000181	0.011260	0.009298
total_day_minutes	-0.022742	0.018833	-0.012776	-0.014438	1.000000	-0.003583	1.000000	-0.004000	-0.006227	-0.001777	-0.030623	0.000553	-0.004012	0.010752	0.013082	0.010726	-0.030671
total_day_calls	0.016283	-0.001316	-0.002650	0.000085	-0.003583	1.000000	-0.003577	-0.003548	-0.011278	-0.002203	0.000605	-0.002700	-0.003545	0.016550	-0.022221	0.016541	0.000625
total_day_charge	-0.022735	0.018839	-0.012783	-0.014445	1.000000	-0.003577	1.000000	-0.003999	-0.006236	-0.001775	-0.030628	0.000549	-0.004011	0.010757	0.013077	0.010731	-0.030675
total_eve_minutes	-0.025878	0.004479	0.020704	0.018572	-0.004000	-0.003548	-0.003999	1.000000	0.000907	0.041175	-0.008282	0.001632	1.000000	-0.021098	0.010719	-0.021125	-0.008345
number_customer_service_calls	0.012635	-0.019765	-0.019309	-0.009096	-0.006227	-0.011278	-0.006236	0.000907	1.000000	-0.013604	-0.032484	0.019849	0.000904	-0.022024	-0.022733	-0.022012	-0.032438
total_intl_calls	0.028990	-0.001989	-0.012405	0.000647	-0.001777	-0.002203	-0.001775	0.041175	-0.013604	1.000000	0.025863	-0.001575	0.041168	-0.021466	0.021105	-0.021428	0.025842
total_intl_charge	0.010828	0.018457	0.010257	0.009337	-0.030623	0.000605	-0.030628	-0.008282	-0.032484	0.025863	1.000000	-0.014016	-0.008270	0.002038	-0.003367	0.002048	0.999993
total_eve_calls	0.005329	-0.012504	-0.007505	-0.004580	0.000553	-0.002700	0.000549	0.001632	0.019849	-0.001575	-0.014016	1.000000	0.001652	0.010906	-0.009442	0.010909	-0.013976
total_eve_charge	-0.025873	0.004482	0.020709	0.018583	-0.004012	-0.003545	-0.004011	1.000000	0.000904	0.041168	-0.008270	0.001652	1.000000	-0.021099	0.010718	-0.021125	-0.008334
total_night_minutes	-0.021610	-0.007317	0.003341	0.011243	0.010752	0.016550	0.010757	-0.021098	-0.022024	-0.021466	0.002038	0.010906	-0.021099	1.000000	0.032737	0.999999	0.002022
total_night_calls	-0.006445	0.013870	0.005222	0.000181	0.013082	-0.022221	0.013077	0.010719	-0.022733	0.021105	-0.003367	-0.009442	0.010718	0.032737	1.000000	0.032720	-0.003280
total_night_charge	-0.021657	-0.007323	0.003361	0.011260	0.010726	0.016541	0.010731	-0.021125	-0.022012	-0.021428	0.002048	0.010909	-0.021125	0.999999	0.032720	1.000000	0.002032
total_intl_minutes	0.010933	0.018446	0.010206	0.009298	-0.030671	0.000625	-0.030675	-0.008345	-0.032438	0.025842	0.999993	-0.013976	-0.008334	0.002022	-0.003280	0.002032	1.000000

Colinearity – Interpretation

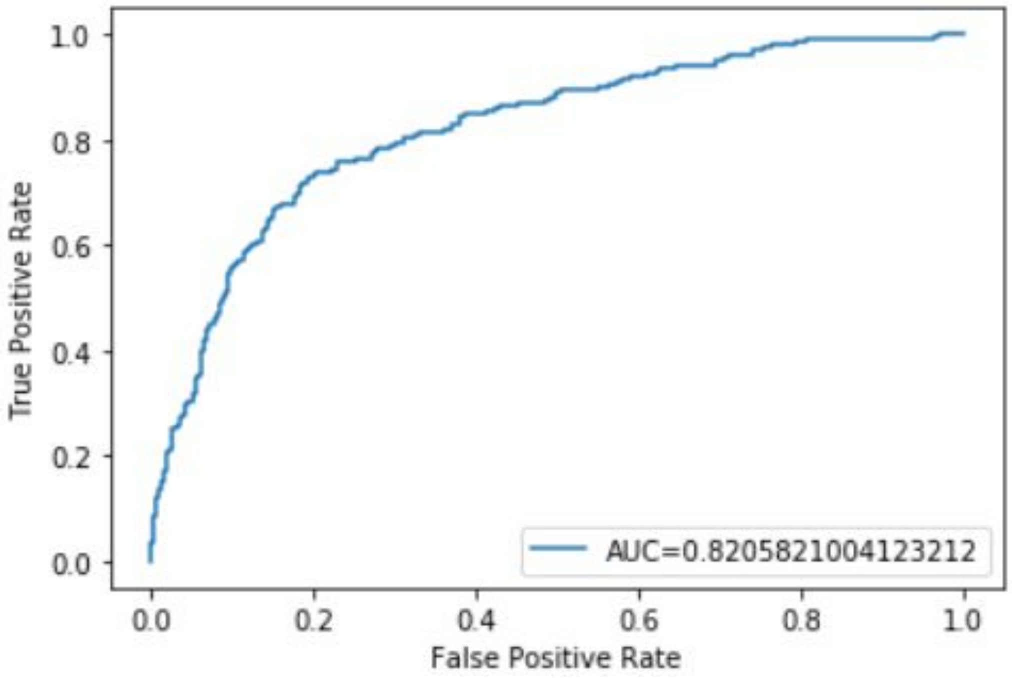
For every client that subscribed to the international plan, the odds for churning are 6 times as large as the odds for not churning when all other variables are held constant.

As variable "total_night_charge" increases by one unit, the odds for churning are over 1,8 as large as the odds for not churning.

	coef
international_plan	6.060882
total_night_charge	1.767848
number_customer_service_calls	1.634395
total_intl_charge	1.209986
total_day_charge	1.084368
total_eve_charge	1.041682
account_length	1.000943
total_day_calls	0.996788
total_night_calls	0.993072
total_eve_calls	0.991382
total_intl_calls	0.918696
voice_mail_plan	0.355987

Colinearity - Evaluation

```
Accuracy:
0.864313725490196
Classification report:
              precision    recall  f1-score   support

           0       0.88      0.98      0.92      1085
           1       0.64      0.20      0.31       190

    accuracy                           0.86      1275
   macro avg       0.76      0.59      0.62      1275
weighted avg       0.84      0.86      0.83      1275

Confusion Matrix
[[1064   21]
 [ 152   38]]
```

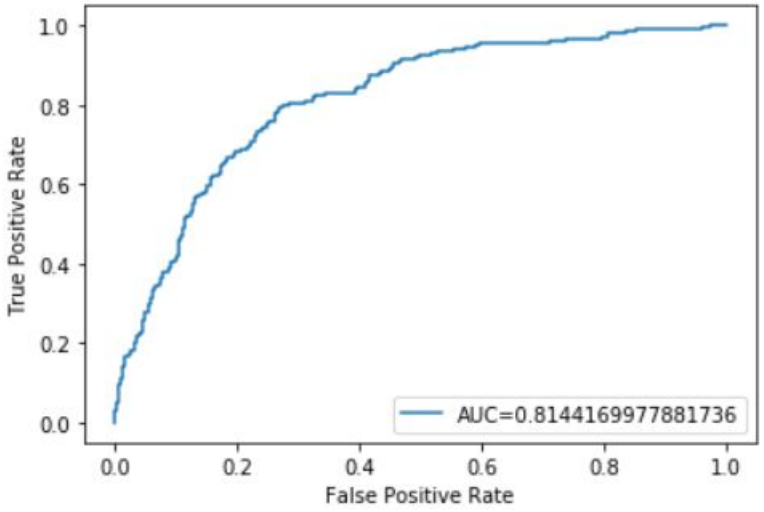


Features Selection - Stepwise

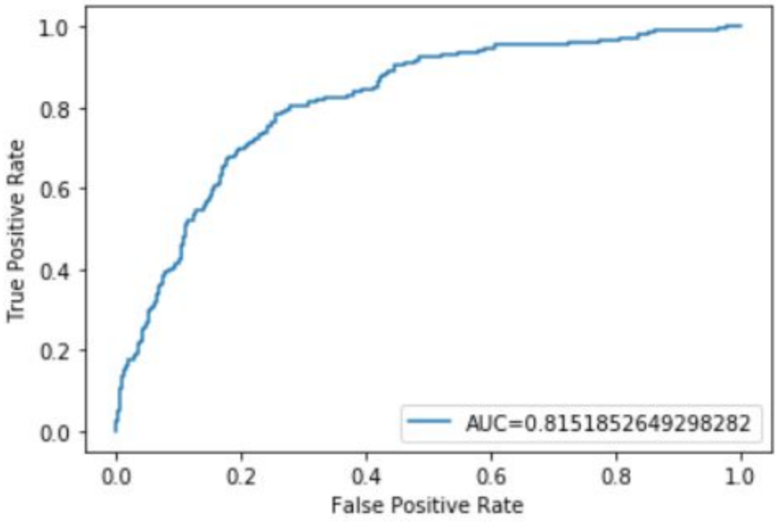
```
from sklearn.feature_selection import SequentialFeatureSelector as SFS

feature_names = np.array(X_train_norm.columns)

sfs1 = SFS(model_lr,
            n_features_to_select=6, direction='forward', scoring="accuracy", cv=5)

sfs1 = sfs1.fit(X_train_norm, y_train)

X_train_norm_sel = X_train_norm[feature_names[sfs1.get_support()]]

model_lr.fit(X_train_norm_sel, y_train)
```

Features Selection - Forward

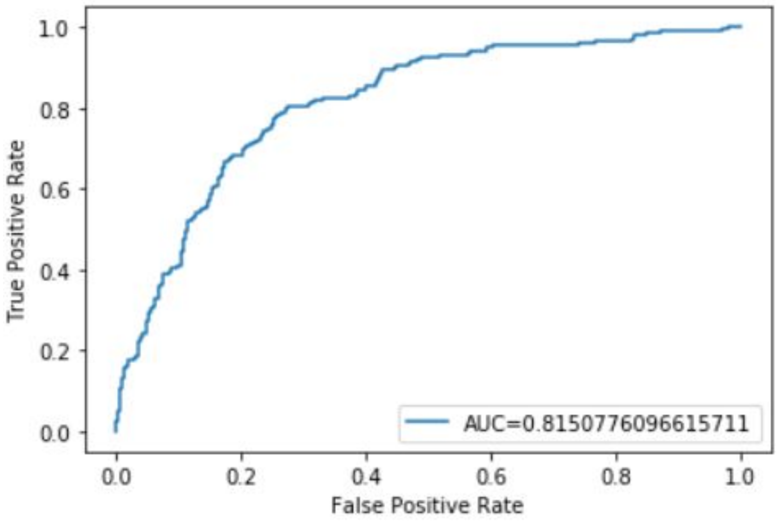
Number of features to select : 5



Number of features to select : 6



Number of features to select : 7



Features Selection - Forward

Number of features to select : 6

```
Accuracy:
0.8619607843137255
Classification report:
              precision    recall  f1-score   support

           0       0.87      0.98      0.92      1087
           1       0.61      0.18      0.27       188

    accuracy                           0.86      1275
   macro avg       0.74      0.58      0.60      1275
weighted avg       0.83      0.86      0.83      1275

Confusion Matrix
[[1066   21]
 [ 155   33]]
```

	coef
international_plan	6.534387
total_night_charge	2.127879
number_customer_service_calls	1.586200
total_day_charge	1.074415
total_eve_charge	1.045447
total_intl_calls	0.912596

For every client that subscribed to the international plan, the odds for churning are 6,5 times as large as the odds for not churning when all other variables are held constant.

As variable “total_night_charge” increases by one unit, the odds for churning are over 2,13 as large as the odds for not churning.

Visualization

Benefits

- Gives more understandable product for the business
- Further analysis of the results
- Analysis of impact

Process

1. Define the goal of the visualization
2. Choose the support
3. Define KPI/Graphics
4. Architecture/Design

Types of Goals

Data Analysis	Model Analysis	Impact Analysis
- Description of the population - Features Analysis	- Process description - Perfomance indicators - Comparison of models	Analysis on the data post-marketing campaign, showing its impact on the churn rate.

Types of Support

Static	Report	Apps
- Excel - Outputs : matplotlib, seaborn..	- Power BI - Tableau - Data Studio	- R Shiny - Django - Flask

Statics Visualizations

Advantages	Disadvantages
- Easy to implement - A lot of solutions - Total flexibility on the graphic design	- Choosing among all solutions - Not dynamic : every parameter changed means a new display

Reports

Advantages	Disadvantages
- Solutions « click-button » - Dynamic : graphics and KPI’s change when user applies filters - Relatively easy to implement (security issues taken care of by the software producer)	- Unflexible : often you can’t create your own graphics. - Special kinds of langages : if you want to unlock the full power of those software, you need to learn their langages (DAX and M for Power BI)

Apps

Advantages	Disadvantages
- Total Flexibility : you can create your own graphics and fully manage the User Experience - Dynamic	- Difficult to implement : security issues appears because Apps are linked to a network. - Special langages : if you want to fully manage your app, you need skills in informatic langages like Java, HTML, etc.

Power BI Example – Data Analysis

Power BI Example – Model Analysis

Power BI Example – Impact Analysis