Statistique Descriptive
Strategies for Imbalanced Data – the churn example

Séverine Affeldt

UFR Mathématiques et Informatique
MLDS - Centre Borelli

Université de Paris

Customer Churn

Churn, attrition, turnover, defection $\Leftrightarrow$ loss of customers (or clients)

Cost of existing customer <u>retention</u> $\ll$ Cost of new customer <u>acquisition</u>

$\Rightarrow$ Customer churn analysis is a key business metric.

Customer churn analysis can help to **win back defecting clients**
and assesses their propensity of risk to churn

Voluntary churn

The costumer decides to switch to another company.

→ Analysis concentrates on this type
→ Factors related to customer-company relationship (eg., billing interactions, after-sales help)

Involuntary churn

The costumer switches to another company due to special circumstances (eg., relocation, death)

→ Usually excluded from churn analysis

Voluntary analysis provides a small prioritized list of potential defectors ⇒ new marketing programs on a subset of customers that are most vulnerable to churn.

telecom-customer-churn-prediction.ipynb

data : churn_prediction_data.csv

We want to identify the variables that **influence the customer churn**.

1. First, load the data and explore the different variables. Where are the variable **types**? Are there **missing data**? Where is the **target variable**?

   1. Data : Loading and manipulating the data

`telecom-customer-churn-prediction.ipynb`

data : `churn_prediction_data.csv`

We want to identify the variables that **influence the customer churn**.

1. First, load the data and explore the different variables. Where are the variable **types**? Are there **missing data**? Where is the **target variable**?

   1. Data : Loading and manipulating the data

2. Choose a plot to display the proportion of customer churn (eg., pie plot). exp.: Pie chart using ggplot2

   2. Exploratory Data Analysis : Customer attrition in data

`telecom-customer-churn-prediction.ipynb`

data : `churn_prediction_data.csv`

We want to identify the variables that **influence the customer churn**.

1. First, load the data and explore the different variables. Where are the variable **types**? Are there **missing data**? Where is the **target variable**?
   1. Data : Loading and manipulating the data
2. Choose a plot to display the proportion of customer churn (eg., pie plot).
   exp.: Pie chart using ggplot2
   2. Exploratory Data Analysis : Customer attrition in data
3. For all **qualitative data**, display, side by side, the proportion of their categories for the churn and non churn population.
   2.2.1 Visualizing churn for qualitative variables

`telecom-customer-churn-prediction.ipynb`

data : `churn_prediction_data.csv`

We want to identify the variables that **influence the customer churn**.

1. First, load the data and explore the different variables. Where are the variable **types**? Are there **missing data**? Where is the **target variable**?
   1. Data : Loading and manipulating the data
2. Choose a plot to display the proportion of customer churn (eg., pie plot).
   exp.: Pie chart using ggplot2
   2. Exploratory Data Analysis : Customer attrition in data
3. For all **qualitative data**, display, side by side, the proportion of their categories for the churn and non churn population.
   2.2.1 Visualizing churn for qualitative variables
4. For all **quantitative data**, display the distribution of values for the churn and non churn population.
   exp.: Side by side histogram using matplotlib
   2.2.2 Visualizing churn for quantitative variables

telecom-customer-churn-prediction.ipynb

data : churn_prediction_data.csv

We want to identify the variables that **influence the customer churn**.

1. First, load the data and explore the different variables. Where are the variable **types**? Are there **missing data**? Where is the **target variable**?
   1. Data : Loading and manipulating the data
2. Choose a plot to display the proportion of customer churn (eg., pie plot).
   exp.: Pie chart using ggplot2
   2. Exploratory Data Analysis : Customer attrition in data
3. For all **qualitative data**, display, side by side, the proportion of their categories for the churn and non churn population.
   2.2.1 Visualizing churn for qualitative variables
4. For all **quantitative data**, display the distribution of values for the churn and non churn population.
   exp.: Side by side histogram using matplotlib
   2.2.2 Visualizing churn for quantitative variables
5. Provide a pairwise scatterplot for all the quantitative variables.
   exp.: Scatterplot matrix
   2.2.3 Visualizing pairwise scatterplot for quantitative variables

`telecom-customer-churn-prediction.ipynb`

data : `churn_prediction_data.csv`

Let's focus on the **tenure** information.

1. Show the churn attrition in tenure groups (eg., side by side histogram)
   2.3.1 Customer attrition in tenure groups

`telecom-customer-churn-prediction.ipynb`

data : `churn_prediction_data.csv`

Let's focus on the **tenure** information.

1. Show the churn attrition in tenure groups (eg., side by side histogram)
   2.3.1 Customer attrition in tenure groups
2. Explore the average charges by tenure group
   2.3.2 Average Charges by tenure groups

evaluating_classification_models.ipynb

Metrics : accuracy, confusion matrix, ROC curve, AUC & Lift

Accuracy

Def.: The percent of cases classified correctly, ie. a measure of total error

$$accuracy = \frac{\sum TruePositive + \sum TrueNegative}{SampleSize}$$

Confusion matrix

Def.: The record counts by predicted and actual classification status

		Predicted response	
		$\hat{y} = 1$	$\hat{y} = 0$
True response	$y = 1$	True Positive (TP)	False Negative (FN)
	$y = 0$	False Positive (FP)	True Negative (TN)

evaluating_classification_models.ipynb

→ *1. Confusion matrix*

The rare case problem

The problem: There is an imbalance between the classes to be predicted. The rare class is typically 1, and misclassifying 1s as 0s is constier than the opposite: correctly identifying a fraud saves you more than identifying a non fraud...

The metric issue: In case you have 99.9% of non fraudalent actions, a very accurate model will be a model that classify everything as 0...which is useless!

Precision, Recall and Specificity

Precision: The percent of predicted 1s that are actually 1s. It is the *accuracy* of a predicted positive outcome.

$$precision = \frac{\sum TruePositive}{\sum TruePositive + \sum FalsePositive}$$

Recall or sensitivity: The percent of all 1s that are correctly classified as 1s. It is the proportion of 1s correctly indentified by the model. In other words, the *recall* measures the strengh of the model to predict a positive outcome.

$$recall = \frac{\sum TruePositive}{\sum TruePositive + \sum FalseNegative}$$

Specificity: It measures the models's ability to predict a negative outcome.

$$specificity = \frac{\sum TrueNegative}{\sum TrueNegative + \sum FalsePositive}$$

Precision, Recall and Specificity and confusion matrix

Predicted response

		$\hat{y} = 1$	$\hat{y} = 0$	
True response	$y = 1$	True Positive	False Negative	Recall $TP/(y = 1)$
	$y = 0$	False Positive	True Negative	Specificity $TN/(y = 0)$
		Precision $TP/(\hat{y} = 1)$		

evaluating_classification_models.ipynb

→ *2. Precision, Recall and Specificity*

Receiver Operating Characteristics

Caution: There is a trade-off between *recall* and *specificity*: the model should be good at classifying 1s, without misclassifying more 0s than 1s! This trade-off is captured by the ROC (Receiver Operating Characteristics) curve, as it plots the *recall* (y-axis) against the *specificity* (x-axis). Each point corresponds to a different cutoff to determine how to classify a record.

Process

1. Sort the record by the predicted probability of being a 1, starting with the most probable
2. Compute the cumulative specificity and recall based on the sorted records

Visual inspection: The diagonal line corresponds to a random classifier. An effective classifier has a ROC in the upper-left corner $\Leftrightarrow$ it correctly identifies lots of 1s without misclassifying lots of 0s as 1s.

`evaluating_classification_models.ipynb`
$\rightarrow$ *3. ROC Curve*

Area Underneath the Curve

Caution: The ROC curve is a graphical tool. The AUC is a metric which corresponds to the area underneath the ROC curve. The larger the value of AUC, the more effective the classifier. An AUC of 1 indicates a perfect classifier: all 1s are correctly classified, and no 0s are misclassified as 1s. A completely ineffective classifier (diagonal line) will have an AUC of 0.5.

`evaluating_classification_models.ipynb`
$\rightarrow$ *4. AUC*