Understanding credit risk

CREDIT RISK MODELING IN PYTHON

Michael Crabtree
Data Scientist, Ford Motor Company

What is credit risk?

- The possibility that someone who has borrowed money will not repay it all
- Calculated risk difference between lending someone money and a government bond
- When someone fails to repay a loan, it is said to be in default
- The likelihood that someone will default on a loan is the probability of default (PD)

What is credit risk?

- The possibility that someone who has borrowed money will not repay it all
- Calculated risk difference between lending someone money and a government bond
- When someone fails to repay a loan, it is said to be in default
- The likelihood that someone will default on a loan is the probability of default (PD)

Payment	Payment Date	Loan Status
$100	Jun 15	Non-Default
$100	Jul 15	Non-Default
$0	Aug 15	Default

Expected loss

- The dollar amount the firm loses as a result of loan default
- Three primary components:
  - Probability of Default (PD)
  - Exposure at Default (EAD)
  - Loss Given Default (LGD)

Formula for expected loss:

```
expected_loss = PD * EAD * LGD
```

Types of data used

Two Primary types of data used:

- Application data
- Behavioral data

Application	Behavioral
Interest Rate	Employment Length
Grade	Historical Default
Amount	Income

Data columns

- Mix of behavioral and application
- Contain columns simulating credit bureau data

Column	Column
Income	Loan grade
Age	Loan amount
Home ownership	Interest rate
Employment length	Loan status
Loan intent	Historical default
Percent Income	Credit history length

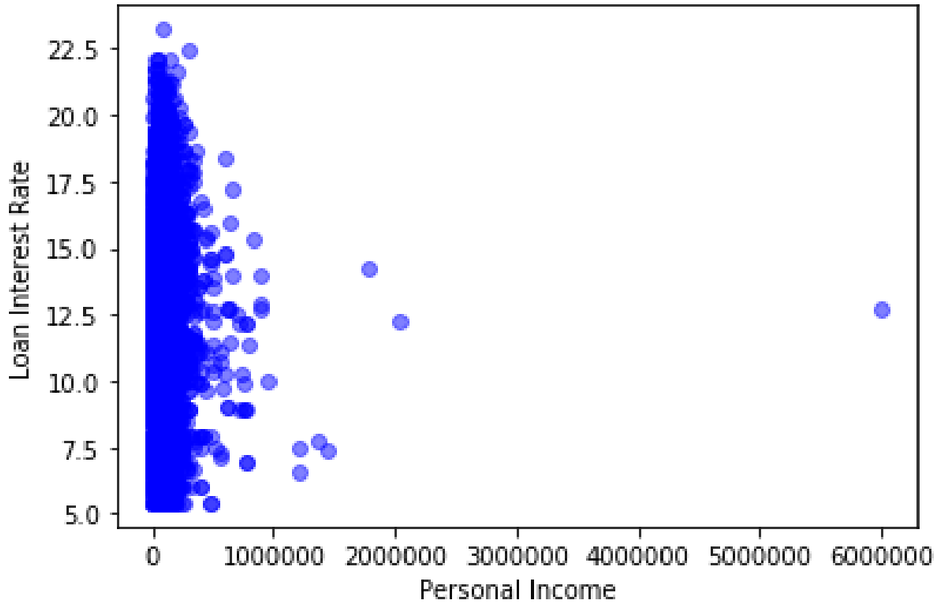
Exploring with cross tables

```
pd.crosstab(cr_loan['person_home_ownership'], cr_loan['loan_status'],
            values=cr_loan['loan_int_rate'], aggfunc='mean').round(2)
```

loan_status	0	1
person_home_ownership		
MORTGAGE	10.06	13.43
OTHER	11.53	13.77
OWN	10.75	12.24
RENT	10.78	13.73

Exploring with visuals

```
plt.scatter(cr_loan['person_income'], cr_loan['loan_int_rate'],c='blue', alpha=0.5)
plt.xlabel("Personal Income")
plt.ylabel("Loan Interest Rate")
plt.show()
```



Let's practice!

CREDIT RISK MODELING IN PYTHON

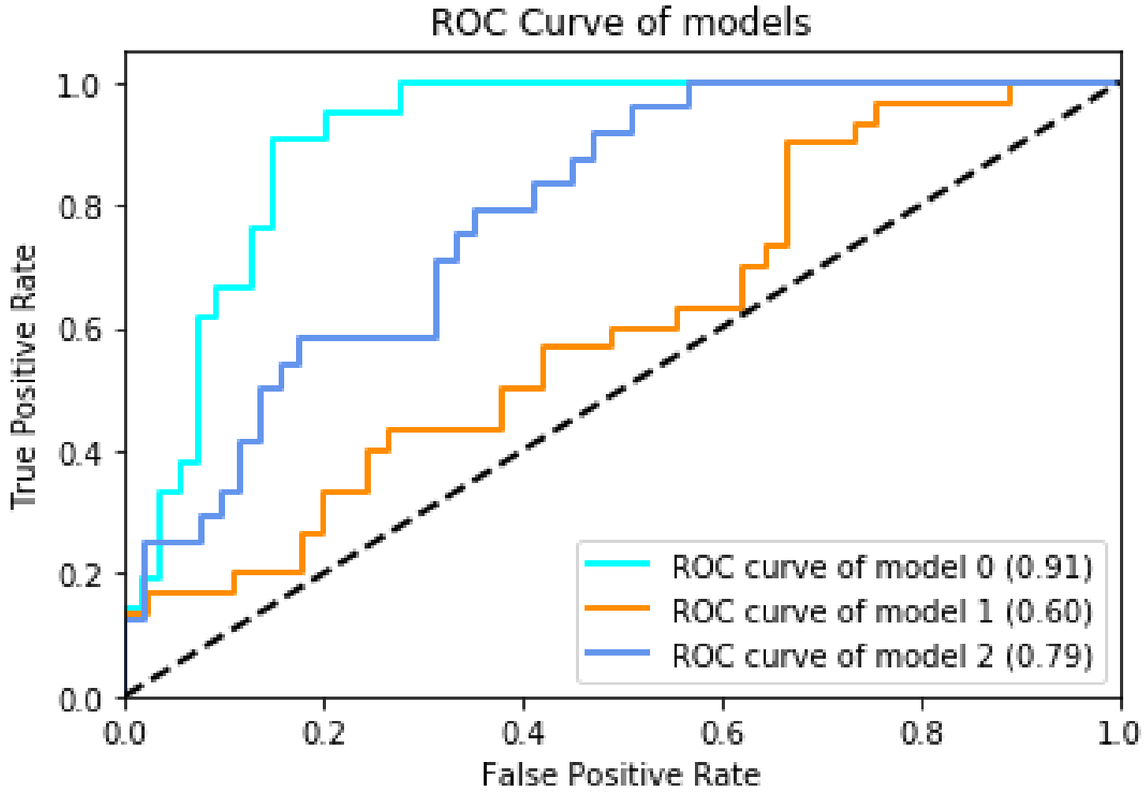
Outliers in Credit Data

CREDIT RISK MODELING IN PYTHON

Michael Crabtree
Data Scientist, Ford Motor Company

Data processing

- Prepared data allows models to train faster
- Often positively impacts model performance



Outliers and performance

Possible causes of outliers:

- Problems with data entry systems (human error)
- Issues with data ingestion tools

Outliers and performance

Possible causes of outliers:

- Problems with data entry systems (human error)
- Issues with data ingestion tools

Feature	Coefficient With Outliers	Coefficient Without Outliers
Interest Rate	0.2	0.01
Employment Length	0.5	0.6
Income	0.6	0.75

Detecting outliers with cross tables

- Use cross tables with aggregate functions

```
pd.crosstab(cr_loan['person_home_ownership'], cr_loan['loan_status'],
            values=cr_loan['loan_int_rate'], aggfunc='mean').round(2)
```

Without Outliers

loan_status	0	1
person_home_ownership		
MORTGAGE	10.06	13.43
OTHER	11.53	13.77
OWN	10.75	12.24
RENT	10.78	13.73

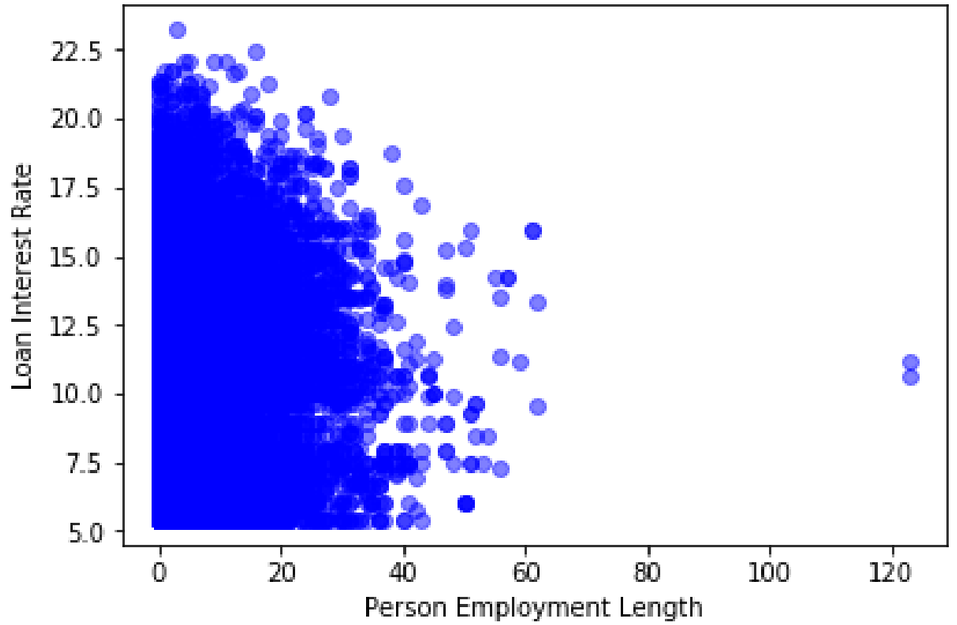
With Outliers

loan_status	0	1
person_home_ownership		
MORTGAGE	10.06	13.43
OTHER	11.53	13.77
OWN	10.75	59183.79
RENT	10.78	13.73

Detecting outliers visually

Detecting outliers visually

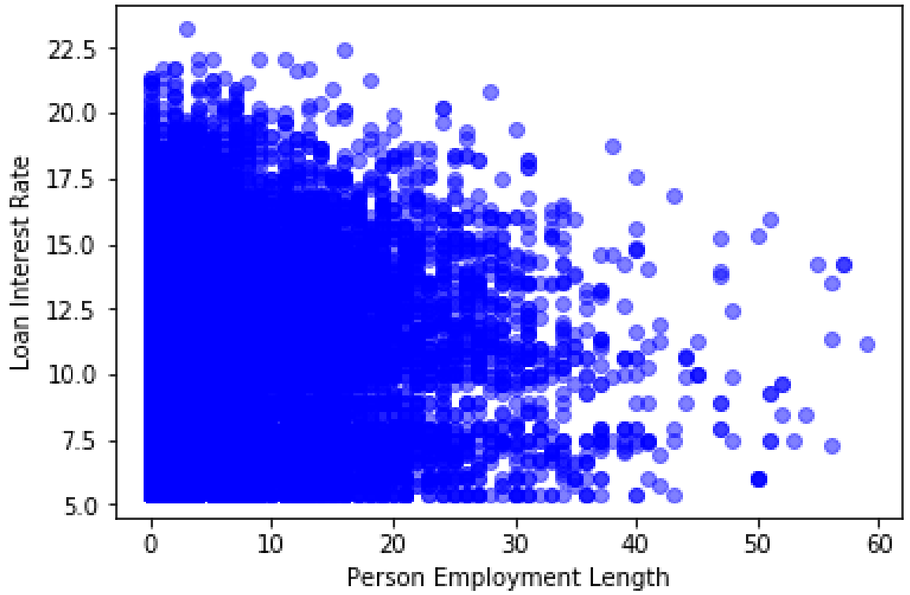
- Histograms
- Scatter plots



Removing outliers

- Use the `.drop()` method within Pandas

```
indices = cr_loan[cr_loan['person_emp_length'] >= 60].index
cr_loan.drop(indices, inplace=True)
```



Let's practice!

CREDIT RISK MODELING IN PYTHON

Risk with missing data in loan data

CREDIT RISK MODELING IN PYTHON

Michael Crabtree
Data Scientist, Ford Motor Company

datacamp

What is missing data?

- NULLs in a row instead of an actual value
- An empty string `''`
- Not an entirely empty row
- Can occur in any column in the data

	person_age	person_income	person_home_ownership	person_emp_length	loan_intent
105	22	12600.0	MORTGAGE	NaN	PERSONAL
222	24	185000.0	MORTGAGE	NaN	EDUCATION
379	24	16800.0	MORTGAGE	NaN	DEBTCONSOLIDATION

Similarities with outliers

- Negatively affect machine learning model performance
- May bias models in unanticipated ways
- May cause errors for some machine learning models

Similarities with outliers

- Negatively affect machine learning model performance
- May bias models in unanticipated ways
- May cause errors for some machine learning models

Missing Data Type	Possible Result
NULL in numeric column	Error
NULL in string column	Error

How to handle missing data

- Generally three ways to handle missing data
  - Replace values where the data is missing
  - Remove the rows containing missing data
  - Leave the rows with missing data unchanged
- Understanding the data determines the course of action

How to handle missing data

- Generally three ways to handle missing data
  - Replace values where the data is missing
  - Remove the rows containing missing data
  - Leave the rows with missing data unchanged
- Understanding the data determines the course of action

Missing Data	Interpretation	Action
NULL in `loan_status`	Loan recently approved	Remove from prediction data
NULL in `person_age`	Age not recorded or disclosed	Replace with median

Finding missing data

- Null values are easily found by using the `isnull()` function
- Null records can easily be counted with the `sum()` function
- `.any()` method checks all columns

```
null_columns = cr_loan.columns[cr_loan.isnull().any()]
cr_loan[null_columns].isnull().sum()
```

```
# Total number of null values per column
person_home_ownership          25
person_emp_length             895
loan_intent                    25
loan_int_rate                3140
cb_person_default_on_file      15
```

Replacing Missing data

- Replace the missing data using methods like `.fillna()` with aggregate functions and methods

```
cr_loan['loan_int_rate'].fillna((cr_loan['loan_int_rate'].mean()), inplace = True)
```

loan_int_rate
5.42
12.42
NaN
10.74
15.27

loan_int_rate
5.420000
12.420000
11.010729
10.740000
15.270000

Dropping missing data

- Uses indices to identify records the same as with outliers
- Remove the records entirely using the `.drop()` method

```
indices = cr_loan[cr_loan['person_emp_length'].isnull()].index
cr_loan.drop(indices, inplace=True)
```

Let's practice!

CREDIT RISK MODELING IN PYTHON