✏️

Chapter 1: Simple Linear Regression Modeling

Created	@September 18, 2024 10:00 PM
Class	Introduction to Regression with statsmodels in Python

Introduction to Regression Analysis

Key Details

- Regression is a statistical tool to analyze relationships between variables
- Course covers linear regression for numeric response variables and logistic regression for logical response variables
- Focus on simple regression with a single explanatory variable
- Uses statsmodels package for insight-focused analysis

1. Fundamentals of Regression

1.1 Example Dataset: Swedish Motor Insurance Claims

- Variables: Number of claims (explanatory) and Total payment (response)
- Data represents different regions in Sweden

1.2 Prerequisites

- Experience with descriptive statistics in pandas
- Understanding of correlation between variables

```
python
Copy
# Example: Calculating mean of variables
df.mean()

# Example: Calculating correlation
df['claims'].corr(df['payment'])
```

1.3 Regression Model Concepts

- Explores relationship between response and explanatory variables
- Allows predictions of response variable based on explanatory variables
- Terms:
    - Response variable (dependent variable, y variable)
    - Explanatory variables (independent variables, x variables)

2. Data Visualization

2.1 Scatter Plots

- Used to visualize relationship between two numeric variables

```
# Example: Creating a scatter plot with seaborn
sns.scatterplot(x='claims', y='payment', data=df)
```

2.2 Adding Trend Lines

- Refines scatter plot by showing overall trend
- Uses linear regression to calculate trend line

```
# Example: Adding a trend line with seaborn
sns.regplot(x='claims', y='payment', data=df, ci=None)
```

3. Course Structure

1. Visualizing and fitting linear regressions
2. Making predictions with linear regressions
3. Quantifying model fit
4. Logistic regression (similar flow as linear regression)

4. Python Packages for Regression

- statsmodels: Optimized for insight (used in this course)
- scikit-learn: Optimized for prediction

Fitting Linear Regression Models

Key Details

- Linear regression trend lines are straight lines
- Defined by intercept and slope
- Uses Ordinary Least Squares (OLS) method
- Implemented using statsmodels library in Python

1. Properties of Straight Lines

1.1 Intercept

- Y-value when x is zero
- Where the line intersects the y-axis

1.2 Slope

- Steepness of the line
- Amount y increases when x increases by one

1.3 Equation of a Straight Line

y = intercept + (slope × x)

2. Estimating Intercept and Slope Visually

2.1 Estimating Intercept

- Look at where trend line intersects y-axis
- Example: Estimated around 20 for Swedish insurance dataset

2.2 Estimating Slope

- Choose two points on the line
- Calculate change in y and change in x
- Divide change in y by change in x
- Example: (400 - 150) / (110 - 40) ≈ 3.5

3. Running Linear Regression in Python

3.1 Using statsmodels

```
from statsmodels.formula.api import ols

# Creating and fitting the model
model = ols(formula="total_payment ~ n_claims", data=df).fit
()

# Viewing model parameters
print(model.params)
```

3.2 Interpreting Results

- Intercept: Close to visual estimate (around 20)
- Slope: 3.4 (slightly lower than visual estimate)

4. Interpreting the Model

- Equation: total_payment = 20 + 3.4 × n_claims
- For each additional claim, total payment increases by 3.4

Key Takeaways

- Linear regression fits a straight line to data
- Can estimate intercept and slope visually
- Use statsmodels.formula.api.ols() to fit regression in Python
- Interpret coefficients as intercept and slope of the line

Linear Regression with Categorical Variables

Key Details

- Explores using categorical explanatory variables in linear regression
- Uses fish market data with species (categorical) and mass (numeric)
- Demonstrates visualization and analysis techniques for categorical data

1. Data Visualization for Categorical Variables

1.1 Histograms for Each Category

```
# Using seaborn's displot for multiple histograms
sns.displot(data=fish_data, x='mass', col='species', col_wrap
=3, bins=9)
```

2. Summary Statistics

2.1 Calculating Mean Mass by Species

```
fish_data.groupby('species')['mass'].mean()
```

3. Running Linear Regression with Categorical Variables

3.1 Basic Linear Regression

```
from statsmodels.formula.api import ols

model = ols(formula="mass ~ species", data=fish_data).fit()
print(model.params)
```

3.2 Interpreting Results

- Intercept represents mean mass of reference category (e.g., bream)
- Other coefficients represent differences from reference category

3.3 Improved Model without Intercept

```
model_no_intercept = ols(formula="mass ~ species + 0", data=f
ish_data).fit()
print(model_no_intercept.params)
```

4. Interpreting Categorical Regression Results

- Coefficients represent mean masses for each species
- For single categorical explanatory variable, coefficients are category means

Key Takeaways

- Use histograms or box plots to visualize categorical vs. numeric data
- Basic linear regression with categories uses one category as reference

- Adding "+ 0" to formula removes intercept, making coefficients direct category means
- With single categorical variable, regression coefficients equal category means