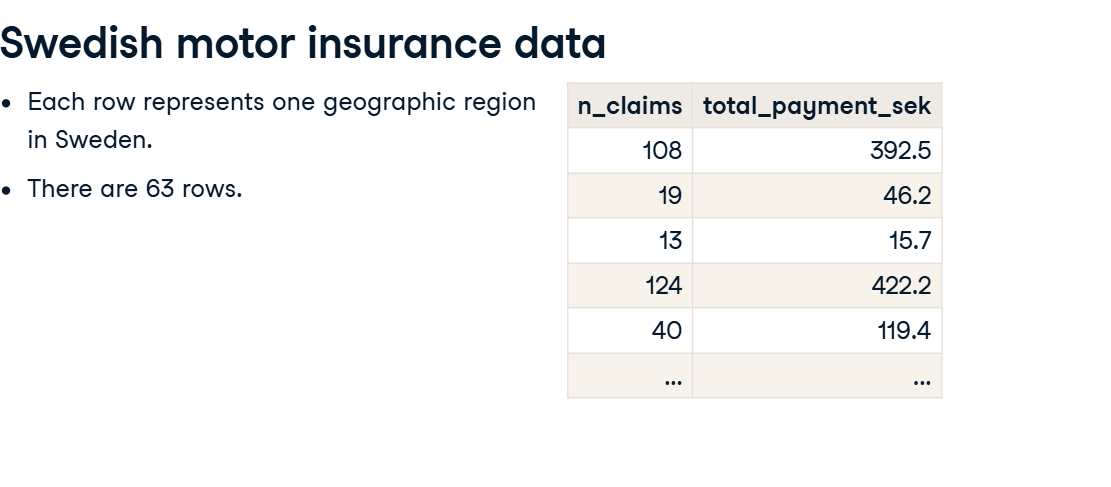
**A tale of two variables**

Hi, my name is Maarten and welcome to the course. You will be learning about regression, a statistical tool to analyze the relationships between variables. Let's start with an example.

**Swedish motor insurance data**

This dataset on Swedish motor insurance claims is as simple as it gets. Each row represents a region in Sweden, and the two variables are the number of claims made in that region, and the total payment made by the insurance company for those claims, in Swedish krona.



**Descriptive statistics**

This course assumes you have experience with calculating descriptive statistics on variables in a DataFrame. For example, calculating the mean of each variable. We can use pandas for this, as shown here. The course also assumes you understand the correlation between two variables. Here, the correlation is 0 point nine one, a strong positive correlation. That means that as the number of claims increases, the total payment typically increases as well.



**What is regression?**

Regression models are a class of statistical models that let you explore the relationship between a response variable and some explanatory variables. That is, given some explanatory variables, you can make predictions about the value of the response variable. In the insurance dataset, if you know the number of claims made in a region, you can predict the amount that the insurance company has to pay out. That lets you do thought experiments like asking how much the company would need to pay if the number of claims increased to two hundred.

**Jargon**

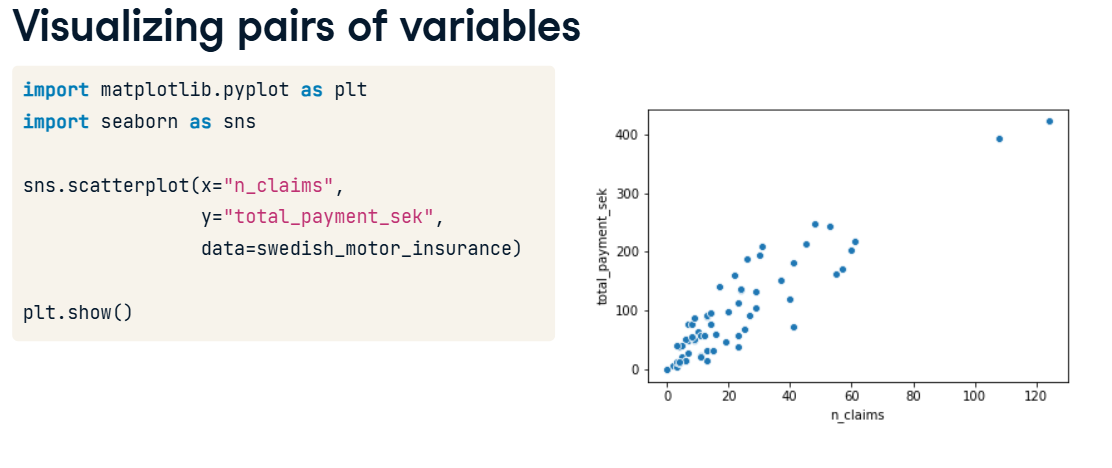
The response variable, the one you want to make predictions on, is also known as the dependent variable or the y variable. These two terms are completely interchangeable. Explanatory variables, used to explain how the predictions will change, are also known as independent variables or x variables. Again, these terms are interchangeable.

**Linear regression and logistic regression**

In this course we're going to look at two types of regression. Linear regression is used when the response variable is numeric, like in the motor insurance dataset. Logistic regression is used when the response variable is logical. That is, it takes True or False values. We'll limit the scope further to only consider simple linear regression and simple logistic regression. This means you only have a single explanatory variable.

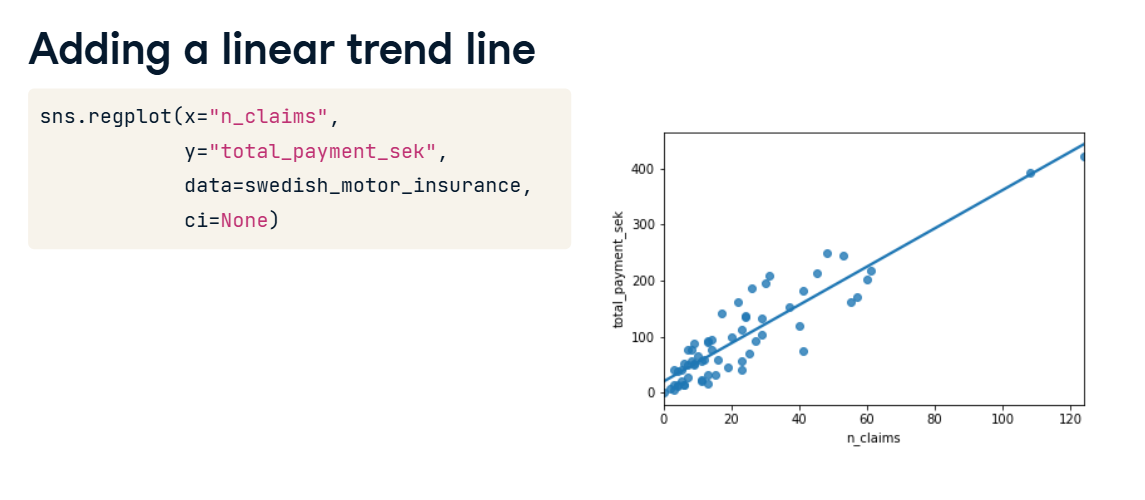
**Visualizing pairs of variables**

Before you start running regression models, it's a good idea to visualize your dataset. To visualize the relationship between two numeric variables, you can use a scatter plot. The course assumes that your data visualization skills are strong enough that you can understand the seaborn code written here. If not, try taking one of DataCamp's courses on seaborn before you begin this course. On the plot, you can see that the total payment increases as the number of claims increases. It would be nice to be able to describe this increase more precisely.



**Adding a linear trend line**

One refinement we can make is to add a trend line to the scatter plot. A trend line means fitting a line that follows the data points. In seaborn, trend lines are drawn using the regplot() function, which adds a trend line calculated using linear regression. By default, regplot() adds a confidence interval around the line, which we can remove by setting the ci argument to None. The trend line is mostly quite close to the data points, so we can say that the linear regression is a reasonable fit.



**Course flow**

Here's the plan for the course. First, we'll visualize and fit linear regressions. Then we'll make predictions with them. Thirdly, we'll look at ways of quantifying whether or not the model is a good fit. In the final chapter, we'll run through this flow again using logistic regression models.

**Python packages for regression**

Before we dive into the first exercise, a word on Python packages for regression. Both statsmodels and scikit-learn can be used. However, statsmodels is more optimized for insight, whereas scikit-learn is more optimized for prediction. Since we'll focus on insight, we'll be using statsmodels in this course.