10 Modeling & SLR

Code

Learning Outcomes

- Understand what models are and how to carry out the four-step modeling process.
- Define the concept of loss and gain familiarity with L_1 and L_2 loss.
- Fit the Simple Linear Regression model using minimization techniques.

Up until this point in the semester, we've focused on analyzing datasets. We've looked into the early stages of the data science lifecycle, focusing on the programming tools, visualization techniques, and data cleaning methods needed for data analysis.

This lecture marks a shift in focus. We will move away from examining datasets to actually *using* our data to better understand the world. Specifically, the next sequence of lectures will explore predictive modeling: generating models to make some predictions about the world around us. In this lecture, we'll introduce the conceptual framework for setting up a modeling task. In the next few lectures, we'll put this framework into practice by implementing various kinds of models.

10.1 What is a Model?

A model is an **idealized representation** of a system. A system is a set of principles or procedures according to which something functions. We live in a world full of systems: the procedure of turning on a light happens according to a specific set of rules dictating the flow of electricity. The truth behind how any event occurs is usually complex, and many times the specifics are unknown. The workings of the world can be viewed as its own giant procedure. Models seek to simplify the world and distill them into workable pieces.

Example: We model the fall of an object on Earth as subject to a constant acceleration of $9.81m/s^2$ due to gravity.

- This is an **approximate** description of a system
- It doesn't account for air resistance, topography, etc
- But in practice, it's **accurate enough** to be useful!

10.1.1 Reasons for Building Models

Why do we want to build models? As far as data scientists and statisticians are concerned, there are three reasons, and each implies a different focus on modeling.

1. **Inference**: Make sense of *phenomena*. For example,

   - How do parents' heights relate to children's heights?
   - What is the correlation of income and education?

   We often want *simple* and *interpretable* models to help us understand relationships

2. **Prediction**: Make accurate predictions about unseen data. Some examples include:

    - Is an email spam or not?
    - Generate a summary of a 10-page long article

    When making prediction, we care more about making extremely *accurate* predictions, at the cost of having a *less interpretable* or *black-box* model. Uninterpretable models are common in fields like deep learning.

3. **Causality**: Assess whether one thing *causes* something else. For example,

    - Does smoking cause lung cancer?
    - Does a job training program increase in employment and wages?

    This is a much harder question! Most statistical tools are designed to infer association, not causation. We will not focus on this task in Data 100, but you can take other advanced classes on causal inference (e.g., Stat 156, Data 102) if you are intrigued!

Most of the time, we aim to strike a balance between building **interpretable** models and building **accurate models**.

Note that these three reasons can overlap! The distinctions are not always clear cut.

10.1.2 [NOT IN SCOPE] Common Types of Models

In general, models can be split into two categories:

1. Deterministic physical (mechanistic) models: Laws that govern how the world works.

    - Kepler's Third Law of Planetary Motion (1619): The ratio of the square of an object's orbital period with the cube of the semi-major axis of its orbit is the same for all objects orbiting the same primary.
        - $T^2 \propto R^3$
    - Newton's Laws: motion and gravitation (1687): Newton's second law of motion models the relationship between the mass of an object and the force required to accelerate it.
        - $F = ma$
        - $F_g = G\frac{m_1 m_2}{r^2}$

2. Probabilistic models: Models that attempt to understand how random processes evolve. These are more general and can be used to describe many phenomena in the real world. These models commonly make simplifying assumptions about the nature of the world.

    - Poisson Process models: Used to model random events that happen with some probability at any point in time and are strictly increasing in count, such as the arrival of customers at a store.

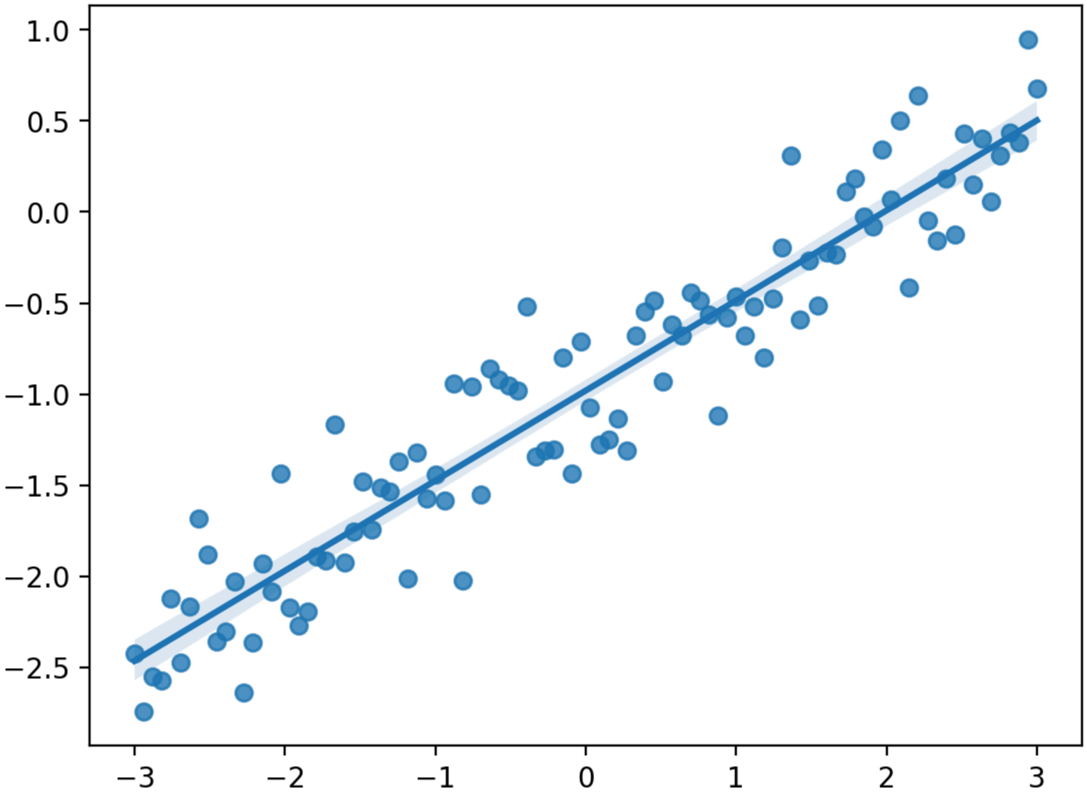
Note: These specific models are not in the scope of Data 100 and exist to serve as motivation.

10.2 Simple Linear Regression

The **regression line** is the unique straight line that minimizes the **mean squared error** of estimation among all straight lines. As with any straight line, it can be defined by a slope and a y-intercept:

- $r = \textbf{correlation}$ between x and y
- $\text{slope} = r \cdot \frac{\text{Standard Deviation of } y}{\text{Standard Deviation of } x}$
- $y\text{-intercept} = \text{average of } y - \text{slope} \cdot \text{average of } x$
- $\text{regression estimate} = y\text{-intercept} + \text{slope} \cdot x$
- $\text{residual} = \text{observed } y - \text{regression estimate}$

▶ Code



10.2.1 Notations and Definitions

For a pair of variables x and y representing our data $\mathcal{D} = \{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$, we denote their means/averages as $\bar{x}$ and $\bar{y}$ and standard deviations as σ_x and σ_y.

10.2.1.1 Standard Units

A variable is represented in standard units if the following are true:

1. 0 in standard units is equal to the mean ($\bar{x}$) in the original variable's units.
2. An increase of 1 standard unit is an increase of 1 standard deviation (σ_x) in the original variable's units.

To convert a variable x_i into standard units, we subtract its mean from it and divide it by its standard deviation. For example, x_i in standard units is $\frac{x_i - \bar{x}}{\sigma_x}$.

10.2.1.2 Correlation

The correlation (r) is the average of the product of x and y, both measured in *standard units*.

In general,

$$r = \frac{1}{n} \sum_{i=1}^{n} (\frac{x_i - \bar{x}}{\sigma_x})(\frac{y_i - \bar{y}}{\sigma_y})$$

However, when $\bar{x} = 0, \bar{y} = 0, \sigma_x = 1$, or $\sigma_y = 1$ (which are all satisfied when x and y are both in standard units),

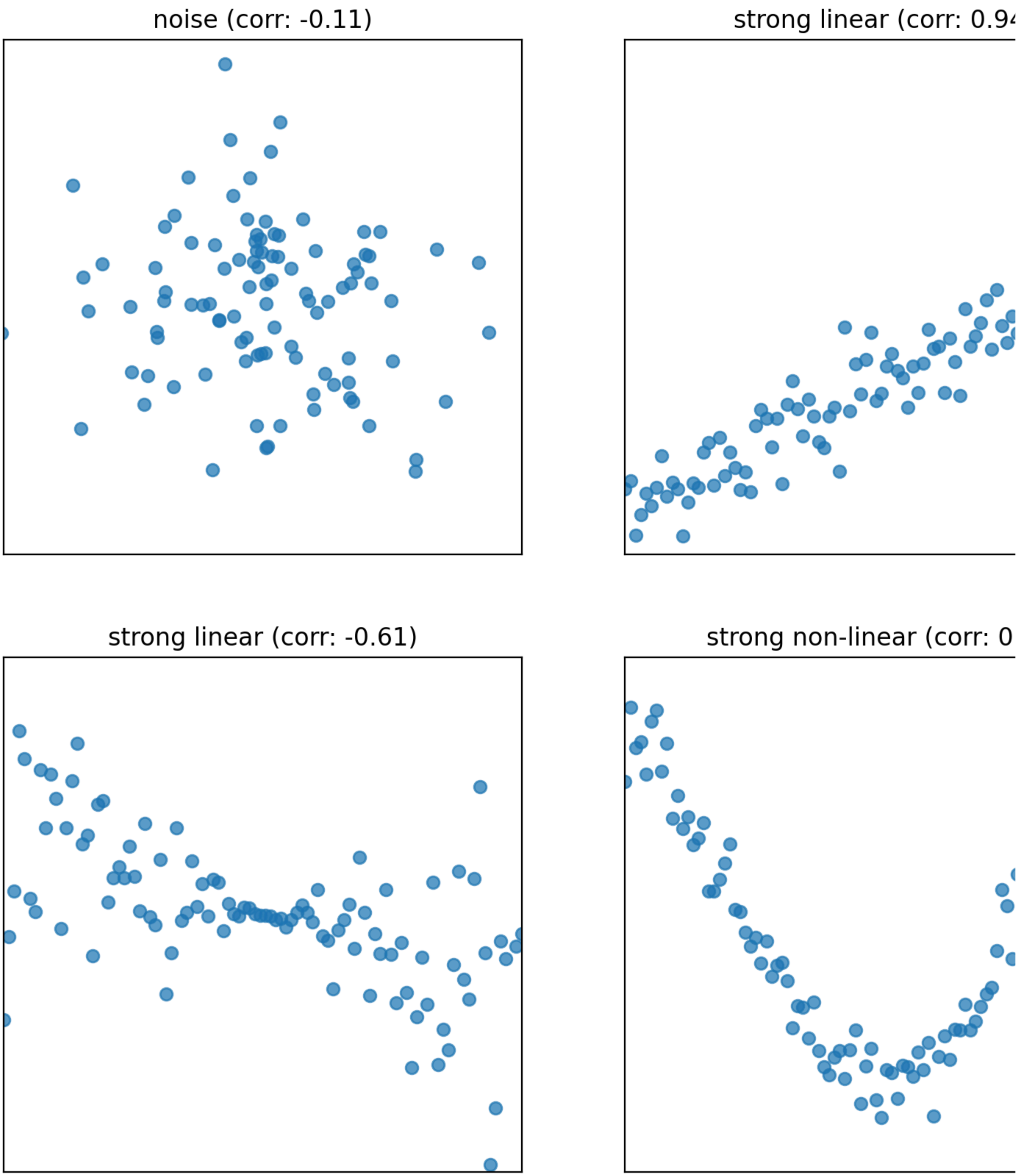
$$r = \frac{1}{n} \sum_{i=1}^{n} x_i y_i$$

This simpler formula is convenient to work with when possible.

1. Correlation measures the strength of a **linear association** between two variables.
2. Correlations range between -1 and 1: $|r| \leq 1$, with $r = 1$ indicating perfect positive linear association, and $r = -1$ indicating perfect negative association. The closer r is to 0, the weaker the linear association is.
3. Correlation says nothing about causation and non-linear association. Correlation does **not** imply causation. When $r = 0$, the two variables are uncorrelated. However, they could still be related through some non-linear relationship.

For an intuitive understanding of correlation, when x_i and y_i have the same sign (in standard units), the (x_i, y_i) pair contributes positively to correlation. *Opposite* signs contrinbute negatively.

▶ Code

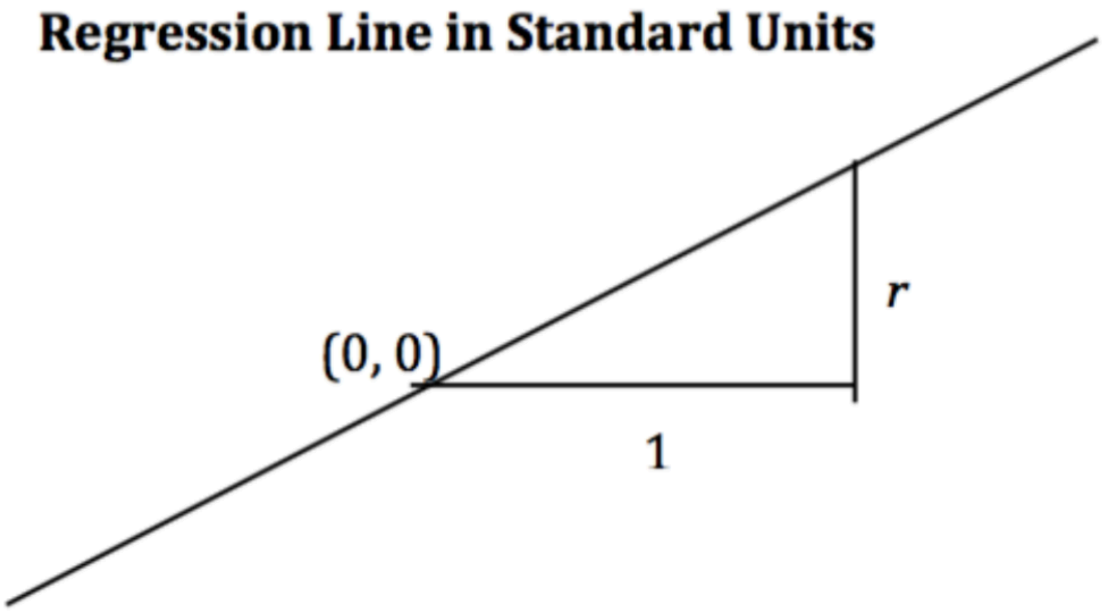


10.2.2 Alternate Form

When the variables y and x are measured in *standard units*, the regression line for predicting y based on x has slope r and passes through the origin.

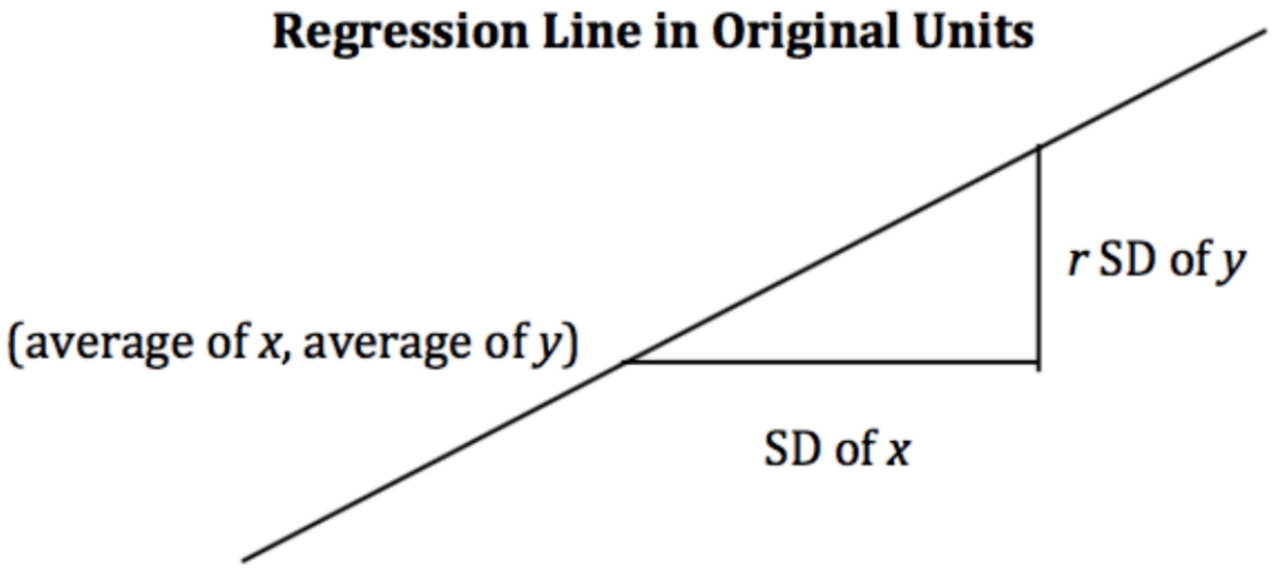
$$\hat{y}_{su} = r \cdot x_{su}$$

Notice that when r = 1, we have perfect prediction!


Regression Line in Standard Units

- In the original units, this becomes

$$\frac{\hat{y}-\bar{y}}{\sigma_y} = r \cdot \frac{x-\bar{x}}{\sigma_x}$$


Regression Line in Original Units

10.2.3 Derivation

Starting from the top, we have our claimed form of the regression line, and we want to show that it is equivalent to the optimal linear regression line: $\hat{y} = \hat{a} + \hat{b}x$.

Recall:

- $\hat{b} = r \cdot \frac{\text{Standard Deviation of } y}{\text{Standard Deviation of } x}$

- $\hat{a} = \text{average of } y - \text{slope} \cdot \text{average of } x$

> Proof:
>
> $$\frac{\hat{y} - \bar{y}}{\sigma_y} = r \cdot \frac{x - \bar{x}}{\sigma_x}$$
>
> Multiply by σ_y, and add $\bar{y}$ on both sides.
>
> $$\hat{y} = \sigma_y \cdot r \cdot \frac{x - \bar{x}}{\sigma_x} + \bar{y}$$
>
> Distribute coefficient $\sigma_y \cdot r$ to the $\frac{x - \bar{x}}{\sigma_x}$ term
>
> $$\hat{y} = (\frac{r\sigma_y}{\sigma_x}) \cdot x + (\bar{y} - (\frac{r\sigma_y}{\sigma_x})\bar{x})$$
>
> We now see that we have a line that matches our claim:
>
> - slope: $r \cdot \frac{\text{SD of y}}{\text{SD of x}} = r \cdot \frac{\sigma_y}{\sigma_x}$
> - intercept: $\bar{y} - \text{slope} \cdot \bar{x}$
>
> Note that the error for the i-th datapoint is: $e_i = y_i - \hat{y}_i$

10.3 The Modeling Process

At a high level, a model is a way of representing a system. In Data 100, we'll treat a model as some mathematical rule we use to describe the relationship between variables.

What variables are we modeling? Typically, we use a subset of the variables in our sample of collected data to model another variable in this data. To put this more formally, say we have the following dataset $\mathcal{D}$:

$$\mathcal{D} = \{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$$

Each pair of values (x_i, y_i) represents a datapoint. In a modeling setting, we call these **observations**. y_i is the dependent variable we are trying to model, also called an **output** or **response**. x_i is the independent variable inputted into the model to make predictions, also known as a **feature**.

Our goal in modeling is to use the observed data $\mathcal{D}$ to predict the output variable y_i. We denote each prediction as $\hat{y}_i$ (read: "y hat sub i").

How do we generate these predictions? Some examples of models we'll encounter in the next few lectures are given below:

$$\hat{y}_i = \theta$$

$$\hat{y}_i = \theta_0 + \theta_1 x_i$$

The examples above are known as **parametric models**. They relate the collected data, x_i, to the prediction we make, $\hat{y}_i$. A few parameters ($\theta, \theta_0, \theta_1$) are used to describe the relationship between x_i and $\hat{y}_i$.

Notice that we don't immediately know the values of these parameters. While the features, x_i, are taken from our observed data, we need to decide what values to give θ, θ_0, and θ_1 ourselves. This is the heart of parametric modeling: *what parameter values should we choose so our model makes the best possible predictions?*

$\hat{\theta}$ is an estimate of a parameter θ based on a sample. The "hat" denotes an estimated or predicted quantity. For example, $\hat{y}$ is a prediction. $\hat{\theta}$ is estimated from data.

Before we move on, note that not all statistical models have parameters! k-Nearest Neighbor classifiers (from Data 8) and KDEs are **non-parametric** models.

To choose our model parameters, we'll work through the **modeling process**.

1. **Choose a model**: How should we represent the world?
2. **Choose a loss function**: How do we quantify prediction error?
3. **Fit the model**: How do we choose the best parameters of our model given our data?
4. **Evaluate model performance**: How do we evaluate whether this process gave rise to a good model?

10.4 Choosing a Model

Our first step is choosing a model: defining the mathematical rule that describes the relationship between the features, x_i, and predictions $\hat{y}_i$.

In Data 8, you learned about the **Simple Linear Regression (SLR) model**. You learned that the model takes the form:

$$\hat{y}_i = a + bx_i$$

In Data 100, we'll use slightly different notation: we will replace a with θ_0 and b with θ_1. This will allow us to use the same notation when we explore more complex models later on in the course.

$$\hat{y}_i = \theta_0 + \theta_1 x_i$$

The parameters of the SLR model are θ_0, also called the intercept term, and θ_1, also called the slope term. To create an effective model, we want to choose values for θ_0 and θ_1 that most accurately predict the output variable. The "best" fitting model parameters are given the special names: $\hat{\theta}_0$ and $\hat{\theta}_1$; they are the specific parameter values that allow our model to generate the best possible predictions.

In Data 8, you learned that the best SLR model parameters are:

$$\hat{\theta}_0 = \bar{y} - \hat{\theta}_1 \bar{x} \qquad \hat{\theta}_1 = r\frac{\sigma_y}{\sigma_x}$$

A quick reminder on notation:

- $\bar{y}$ and $\bar{x}$ indicate the mean value of y and x, respectively
- σ_y and σ_x indicate the standard deviations of y and x
- r is the correlation coefficient, defined as the average of the product of x and y measured in standard units: $\frac{1}{n}\sum_{i=1}^{n}(\frac{x_i-\bar{x}}{\sigma_x})(\frac{y_i-\bar{y}}{\sigma_y})$

In Data 100, we want to understand *how* to derive these best model coefficients. To do so, we'll introduce the concept of a loss function.

10.5 Choosing a Loss Function

We've talked about the idea of creating the "best" possible predictions. This begs the question: how do we decide how "good" or "bad" our model's predictions are?

A **loss function** characterizes the cost, error, or fit resulting from a particular choice of model or model parameters. This function, $L(y, \hat{y})$, quantifies how "bad" or "far off" a single prediction by our model is from a true, observed value in our collected data.

The choice of loss function for a particular model will affect the accuracy and computational cost of estimation, and it'll also depend on the estimation task at hand. For example,

- Are outputs quantitative or qualitative?
- Do outliers matter?
- Are all errors equally costly? (e.g., a false negative on a cancer test is arguably more dangerous than a false positive)

Regardless of the specific function used, a loss function should follow two basic principles:

- If the prediction $\hat{y}_i$ is *close* to the actual value y_i, loss should be low.
- If the prediction $\hat{y}_i$ is *far* from the actual value y_i, loss should be high.

Two common choices of loss function are squared loss and absolute loss.

Squared loss, also known as **L2 loss**, computes loss as the square of the difference between the observed y_i and predicted $\hat{y}_i$:

$$L(y_i, \hat{y}_i) = (y_i - \hat{y}_i)^2$$

Absolute loss, also known as **L1 loss**, computes loss as the absolute difference between the observed y_i and predicted $\hat{y}_i$:

$$L(y_i, \hat{y}_i) = |y_i - \hat{y}_i|$$

L1 and L2 loss give us a tool for quantifying our model's performance on a single data point. This is a good start, but ideally, we want to understand how our model performs across our *entire* dataset. A natural way to do this is to compute the average loss across all data points in the dataset. This is known as the **cost function**, $\hat{R}(\theta)$:

$$\hat{R}(\theta) = \frac{1}{n} \sum_{i=1}^{n} L(y_i, \hat{y}_i)$$

The cost function has many names in the statistics literature. You may also encounter the terms:

- Empirical risk (this is why we give the cost function the name R)
- Error function

- Average loss

We can substitute our L1 and L2 loss into the cost function definition. The **Mean Squared Error (MSE)** is the average squared loss across a dataset:

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

The **Mean Absolute Error (MAE)** is the average absolute loss across a dataset:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

10.6 Fitting the Model

Now that we've established the concept of a loss function, we can return to our original goal of choosing model parameters. Specifically, we want to choose the best set of model parameters that will minimize the model's cost on our dataset. This process is called fitting the model.

We know from calculus that a function is minimized when (1) its first derivative is equal to zero and (2) its second derivative is positive. We often call the function being minimized the **objective function** (our objective is to find its minimum).

To find the optimal model parameter, we:

1. Take the derivative of the cost function with respect to that parameter
2. Set the derivative equal to 0
3. Solve for the parameter

We repeat this process for each parameter present in the model. For now, we'll disregard the second derivative condition.

To help us make sense of this process, let's put it into action by deriving the optimal model parameters for simple linear regression using the mean squared error as our cost function. Remember: although the notation may look tricky, all we are doing is following the three steps above!

Step 1: take the derivative of the cost function with respect to each model parameter. We substitute the SLR model, $\hat{y}_i = \theta_0 + \theta_1 x_i$, into the definition of MSE above and differentiate with respect to θ_0 and θ_1.

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 = \frac{1}{n}\sum_{i=1}^{n}(y_i - \theta_0 - \theta_1 x_i)^2$$

$$\frac{\partial}{\partial \theta_0}\text{MSE} = \frac{-2}{n}\sum_{i=1}^{n} y_i - \theta_0 - \theta_1 x_i$$

$$\frac{\partial}{\partial \theta_1}\text{MSE} = \frac{-2}{n}\sum_{i=1}^{n}(y_i - \theta_0 - \theta_1 x_i)x_i$$

Let’s walk through these derivations in more depth, starting with the derivative of MSE with respect to θ_0.

Given our MSE above, we know that:

$$\frac{\partial}{\partial \theta_0}\text{MSE} = \frac{\partial}{\partial \theta_0}\frac{1}{n}\sum_{i=1}^{n}\left(y_i - \theta_0 - \theta_1 x_i\right)^2$$

Noting that the derivative of sum is equivalent to the sum of derivatives, this then becomes:

$$= \frac{1}{n}\sum_{i=1}^{n}\frac{\partial}{\partial \theta_0}\left(y_i - \theta_0 - \theta_1 x_i\right)^2$$

We can then apply the chain rule.

$$= \frac{1}{n}\sum_{i=1}^{n}2 \cdot \left(y_i - \theta_0 - \theta_1 x_i\right)\dot{(} - 1)$$

Finally, we can simplify the constants, leaving us with our answer.

$$\frac{\partial}{\partial \theta_0}\text{MSE} = \frac{-2}{n}\sum_{i=1}^{n}\left(y_i - \theta_0 - \theta_1 x_i\right)$$

Following the same procedure, we can take the derivative of MSE with respect to θ_1.

$$\frac{\partial}{\partial \theta_1}\text{MSE} = \frac{\partial}{\partial \theta_1}\frac{1}{n}\sum_{i=1}^{n}\left(y_i - \theta_0 - \theta_1 x_i\right)^2$$

$$= \frac{1}{n}\sum_{i=1}^{n}\frac{\partial}{\partial \theta_1}\left(y_i - \theta_0 - \theta_1 x_i\right)^2$$

$$= \frac{1}{n}\sum_{i=1}^{n}2(y_i - \theta_0 \dot{-} \theta_1 x_i)\dot{(} - x_i)$$

$$= \frac{-2}{n}\sum_{i=1}^{n}\left(y_i - \theta_0 - \theta_1 x_i\right)x_i$$

Step 2: set the derivatives equal to 0. After simplifying terms, this produces two **estimating equations**. The best set of model parameters $(\hat{\theta}_0, \hat{\theta}_1)$ *must* satisfy these two optimality conditions.

$$0 = \frac{-2}{n}\sum_{i=1}^{n} y_i - \hat{\theta}_0 - \hat{\theta}_1 x_i \Longleftrightarrow \frac{1}{n}\sum_{i=1}^{n} y_i - \hat{y}_i = 0$$

$$0 = \frac{-2}{n}\sum_{i=1}^{n}(y_i - \hat{\theta}_0 - \hat{\theta}_1 x_i)x_i \Longleftrightarrow \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)x_i = 0$$

Step 3: solve the estimating equations to compute estimates for $\hat{\theta}_0$ and $\hat{\theta}_1$.

Taking the first equation gives the estimate of $\hat{\theta}_0$:

$$\frac{1}{n}\sum_{i=1}^{n} y_i - \hat{\theta}_0 - \hat{\theta}_1 x_i = 0$$

$$\left(\frac{1}{n}\sum_{i=1}^{n} y_i\right) - \hat{\theta}_0 - \hat{\theta}_1\left(\frac{1}{n}\sum_{i=1}^{n} x_i\right) = 0$$

$$\hat{\theta}_0 = \bar{y} - \hat{\theta}_1 \bar{x}$$

With a bit more maneuvering, the second equation gives the estimate of $\hat{\theta}_1$. Start by multiplying the first estimating equation by $\bar{x}$, then subtracting the result from the second estimating equation.

$$\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)x_i - \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)\bar{x} = 0$$

$$\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)(x_i - \bar{x}) = 0$$

Next, plug in $\hat{y}_i = \hat{\theta}_0 + \hat{\theta}_1 x_i = \bar{y} + \hat{\theta}_1(x_i - \bar{x})$:

$$\frac{1}{n}\sum_{i=1}^{n}(y_i - \bar{y} - \hat{\theta}_1(x - \bar{x}))(x_i - \bar{x}) = 0$$

$$\frac{1}{n}\sum_{i=1}^{n}(y_i - \bar{y})(x_i - \bar{x}) = \hat{\theta}_1 \times \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2$$

By using the definition of correlation $\left(r = \frac{1}{n}\sum_{i=1}^{n}(\frac{x_i - \bar{x}}{\sigma_x})(\frac{y_i - \bar{y}}{\sigma_y})\right)$ and standard deviation $\left(\sigma_x = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2}\right)$, we can conclude:

$$r\sigma_x\sigma_y = \hat{\theta}_1 \times \sigma_x^2$$

$$\hat{\theta}_1 = r\frac{\sigma_y}{\sigma_x}$$

Just as was given in Data 8!

Remember, this derivation found the optimal model parameters for SLR when using the MSE cost function. If we had used a different model or different loss function, we likely would have found different values for the best model parameters. However, regardless of the model and loss used, we can *always* follow these three steps to fit the model.

10.7 Evaluating the SLR Model

Now that we've explored the mathematics behind (1) choosing a model, (2) choosing a loss function, and (3) fitting the model, we're left with one final question – how "good" are the predictions made by this "best" fitted model? To determine this, we can:

1. Visualize data and compute statistics:
   - Plot the original data.
   - Compute each column's mean and standard deviation. If the mean and standard deviation of our predictions are close to those of the original observed y_i's, we might be inclined to say that our model has done well.
   - (If we're fitting a linear model) Compute the correlation r. A large magnitude for the correlation coefficient between the feature and response variables could also indicate that our model has done well.
2. Performance metrics:
   - We can take the **Root Mean Squared Error (RMSE)**.
     - It's the square root of the mean squared error (MSE), which is the average loss that we've been minimizing to determine optimal model parameters.
     - RMSE is in the same units as y.
     - A lower RMSE indicates more "accurate" predictions, as we have a lower "average loss" across the data.

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

3. Visualization:
   - Look at the residual plot of $e_i = y_i - \hat{y_i}$ to visualize the difference between actual and predicted values. The good residual plot should not show any pattern between input/features x_i and residual values e_i.

To illustrate this process, let's take a look at **Anscombe's quartet**.

10.7.1 Four Mysterious Datasets (Anscombe's quartet)

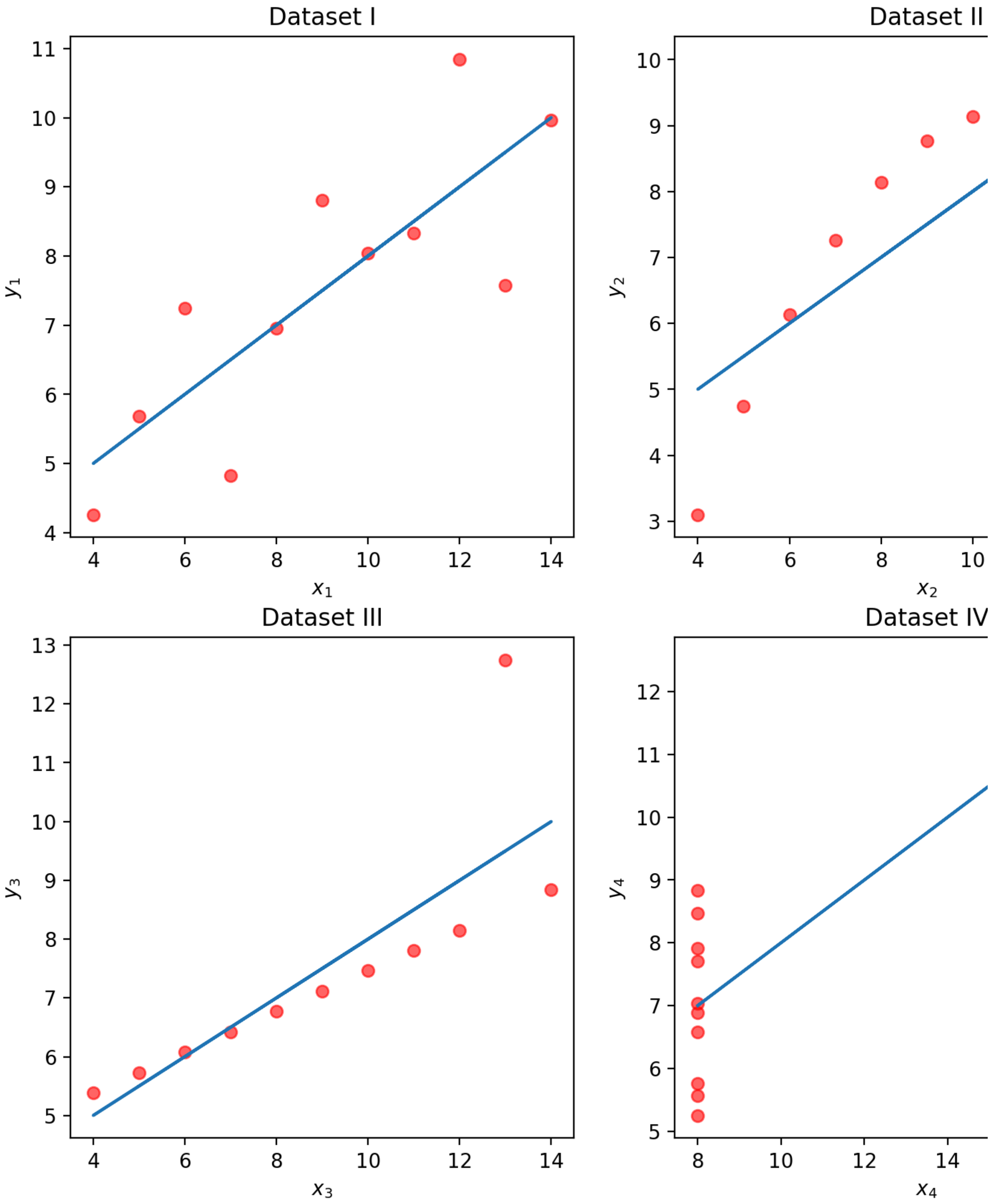
Let's take a look at four different datasets.

▶ Code

▶ Code

▶ Code

▶ Code



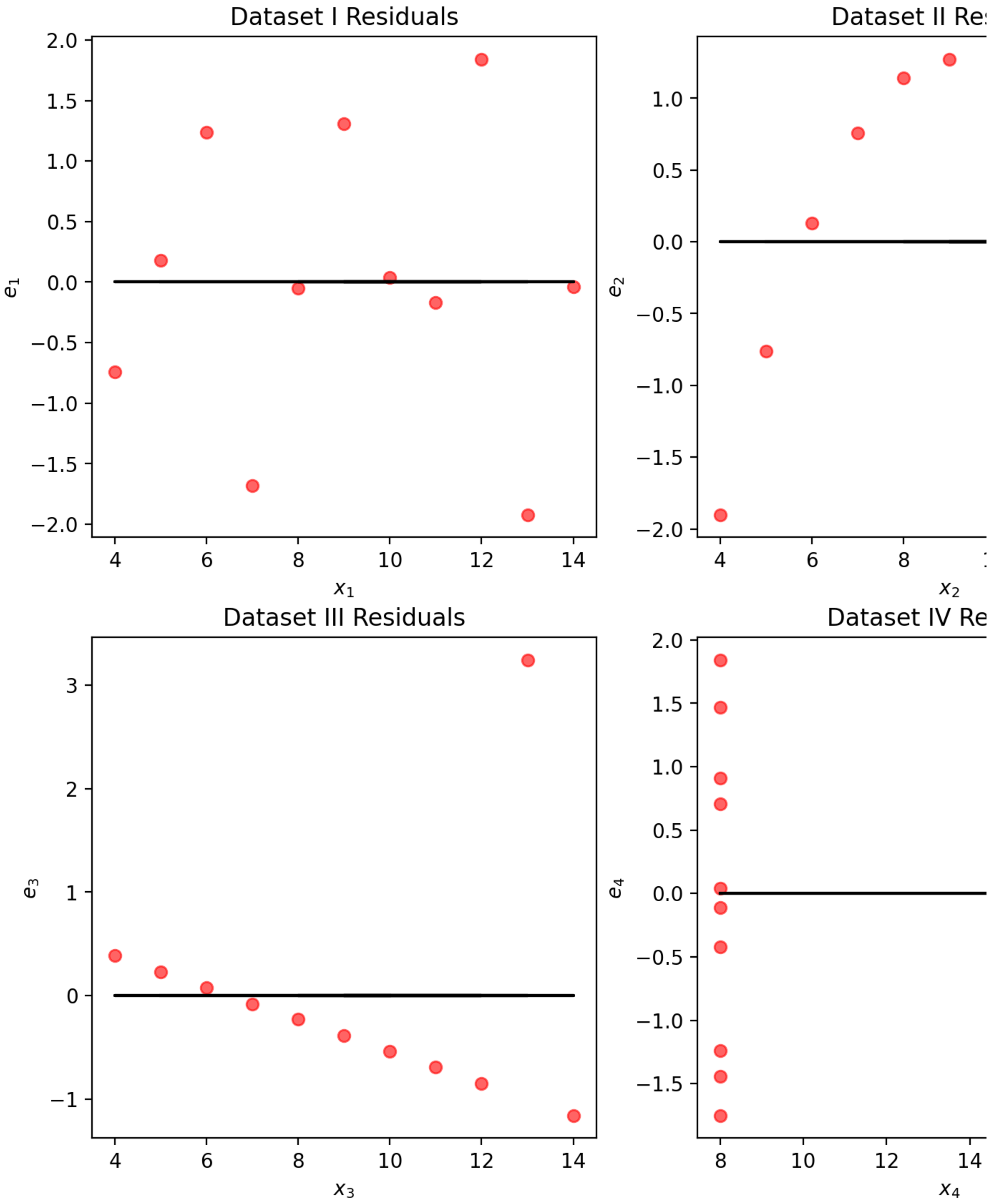
While these four sets of datapoints look very different, they actually all have identical means $\bar{x}, \bar{y}$, standard deviations σ_x, σ_y, correlation r, and RMSE! If we only look at these statistics, we would probably be inclined to say that these datasets are similar.

▶ Code

```
>>> Dataset I:
x_mean : 9.00, y_mean : 7.50
x_stdev: 3.16, y_stdev: 1.94
r = Correlation(x, y): 0.816
    heta_0: 3.00,   heta_1: 0.50
RMSE: 1.119


>>> Dataset II:
x_mean : 9.00, y_mean : 7.50
x_stdev: 3.16, y_stdev: 1.94
r = Correlation(x, y): 0.816
    heta_0: 3.00,   heta_1: 0.50
RMSE: 1.119


>>> Dataset III:
x_mean : 9.00, y_mean : 7.50
x_stdev: 3.16, y_stdev: 1.94
r = Correlation(x, y): 0.816
    heta_0: 3.00,   heta_1: 0.50
RMSE: 1.118


>>> Dataset IV:
x_mean : 9.00, y_mean : 7.50
x_stdev: 3.16, y_stdev: 1.94
r = Correlation(x, y): 0.817
    heta_0: 3.00,   heta_1: 0.50
RMSE: 1.118
```

We may also wish to visualize the model's **residuals**, defined as the difference between the observed and predicted y_i value ($e_i = y_i - \hat{y}_i$). This gives a high-level view of how "off" each prediction is from the true observed value. Recall that you explored this concept in Data 8: a good regression fit should display no clear pattern in its plot of residuals. The residual plots for Anscombe's quartet are displayed below. Note how only the first plot shows no clear pattern to the magnitude of residuals. This is an indication that SLR is not the best choice of model for the remaining three sets of points.

▶ Code


Dataset I Residuals
e_1
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
4
6
8
10
12
14
x_1
Dataset II Res
e_2
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
4
6
8
x_2
Dataset III Residuals
e_3
3
2
1
0
−1
4
6
8
10
12
14
x_3
Dataset IV Re
e_4
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
8
10
12
14
x_4