1 More on Linear Models

We left off last lecture talking about linear models, specifically linear regressions. The general form is

$$X\vec{\beta} = \vec{y}$$

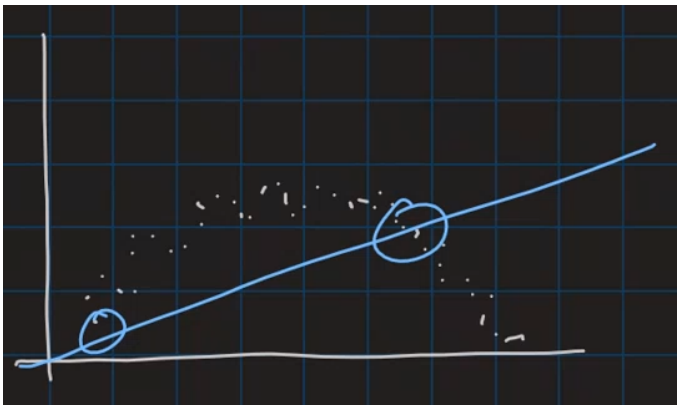
Where X is our design matrix ([1 and x–values]) and $\vec{y}$ is our observed data ([y–values])

Easier Computation

If the columns of X are orthogonal, the least–squares solution is the projection coefficients.

1. Compute $\bar{x}$ — the mean x–value in data ($\frac{\sum x}{n}$)
2. Compute $x* = x - \bar{x}$ — which is the original points minus the mean. These will be orthogonal to x. This is called the mean–deviation form.

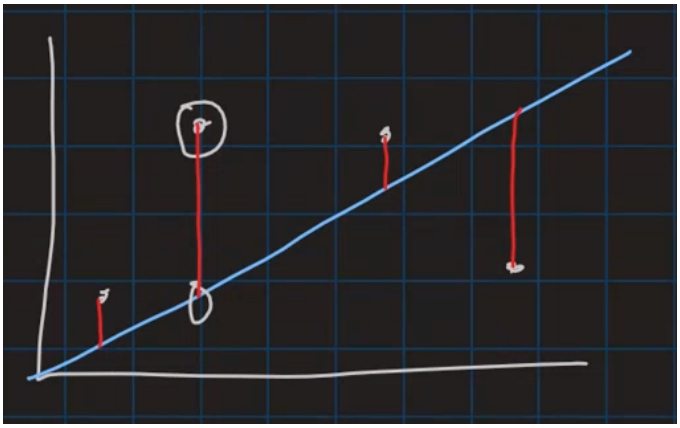
We may need to ask ourselves, how well is this model performing? We may have a spread of data like this where the model will perform decently well in some specific areas but very poorly in others



We can calculate the **residual**

$$\vec{\epsilon} = \vec{y} - X\vec{\beta}$$

Which is the actual data minus the prediction.



This brings us to a **general linear model**

$$\vec{y} = X\vec{\beta} + \vec{\epsilon}$$

The goal is to optimize (minimize) the residual $\vec{\epsilon}$.

To get a model that fits more, we could use any arbitrary function instead of a line

$$y = \beta_0 f_0(x) + \cdots + \beta_n f_n(x)$$

Let's use $\mathbb{P}$ – polynomials

For $\mathbb{P}_2$, $\quad y = \beta_0 + \beta_1 x + \beta_2 x^2$

If we have data in the form (x_i, y_i)

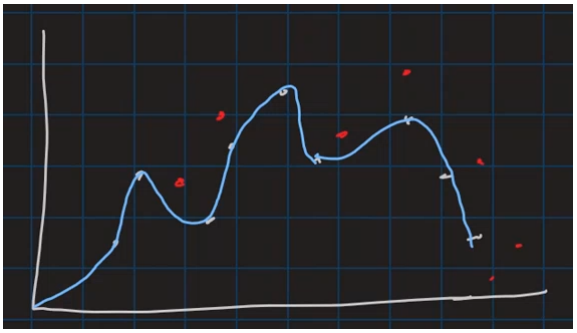
$$\begin{aligned} y_1 &= \beta_0 + \beta_1 x + \beta_2 x^2 + \epsilon_1 \\ &\vdots \\ y_n &= \beta_0 + \beta_1 x + \beta_2 x^2 + \epsilon_n \end{aligned}$$

Written as

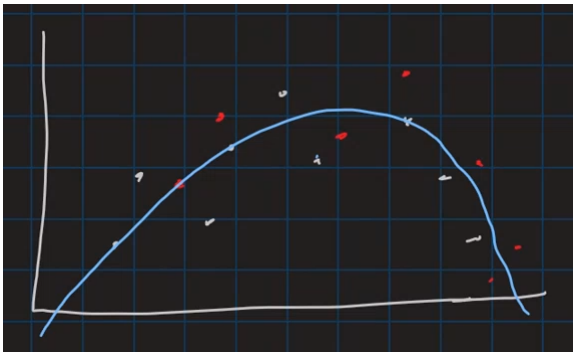
$$\vec{y} = \begin{bmatrix} 1 & x_i & x_i^2 \\ \vdots & \vdots & \vdots \\ 1 & x_n & x_n^2 \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{bmatrix} + \begin{bmatrix} \epsilon_1 \\ \vdots \\ \epsilon_n \end{bmatrix}$$

This allows us to use 2nd degree polynomials to model data. We could also just add more parameters to use higher degree polynomials. We could do that, but then we overfit. Overfitting is when our approximation matches every data point exactly. This does not generalize well to new data. We would have to drastically change the model every time a new data point is added.

We could use two sets of data: our training data (white) and test data (red). If we over–parameterize and overfit our data, it matches the training data exactly, and $\epsilon_{tr} = 0$. But then when we move to test data, it doesn't fit well at all. We get $\epsilon_{te} >> 0$



Here, we use a model that is not overfitted. It matches the training data pretty well, and it also matches the test data pretty well with no change.

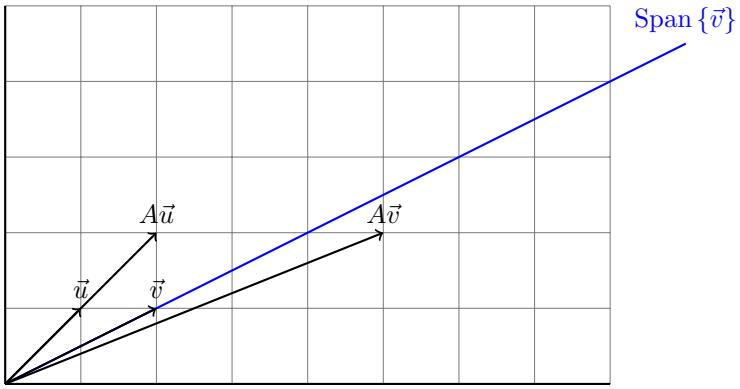


2 Eigenvectors and Eigenvalues

Eigenvectors and Eigenvalues can tell us what the most important dimensions of our data set is. We'll start with an example.

Let $A = \begin{bmatrix} 3 & -1 \\ 0 & 2 \end{bmatrix}$, $\vec{u} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$, $\vec{v} = \begin{bmatrix} 2 \\ 1 \end{bmatrix}$. Think of A as a linear transformation. What happens when we transform each vector with A?

$$A\vec{u} = \begin{bmatrix} 2 \\ 2 \end{bmatrix} \qquad A\vec{v} = \begin{bmatrix} 5 \\ 2 \end{bmatrix}$$



Notice that $\vec{v}$ was knocked off of the Span $\{\vec{v}\}$, but $\vec{u}$ wasn't knocked off Span $\{\vec{u}\}$. This means that $A\vec{u} = \lambda\vec{u}$. This scalar value λ is an **eigenvalue** and the vector $\vec{u}$ is an **eigenvector**.

A scalar λ is an eigenvalue of A is there is a nontrivial solution $\vec{x}$ of $A\vec{x} = \lambda\vec{x}$.

$$A\vec{x} - \lambda\vec{x} = \vec{0}$$

λ can be turned into a matrix by making a diagonal matrix with all values on the diagonal λ and all other values 0. This lets us say

$$(A - \lambda I)\vec{x} = \vec{0}$$

By definition, an eigenvector is nonzero. This implies we need a nontrivial solution to $(A - \lambda I)\vec{x} = \vec{0}$.

Example

Show that 2 is an eigenvalue of A.

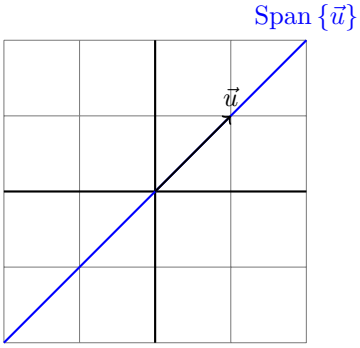
$A = \begin{bmatrix} 3 & -1 \\ 0 & 2 \end{bmatrix}$, solve $A\vec{x} = 2\vec{x}$.

$$A - 2I = \begin{bmatrix} 3 & -1 \\ 0 & 2 \end{bmatrix} - \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} = \begin{bmatrix} 1 & -1 \\ 0 & 0 \end{bmatrix}$$

We can see that this is clearly not a trivial solution (it's not $\vec{0}$), and the columns are linearly dependent. What is the eigenvector? We would normally take that resulting matrix above and set up an augmented matrix with $\vec{0}$, but it's already reduced.

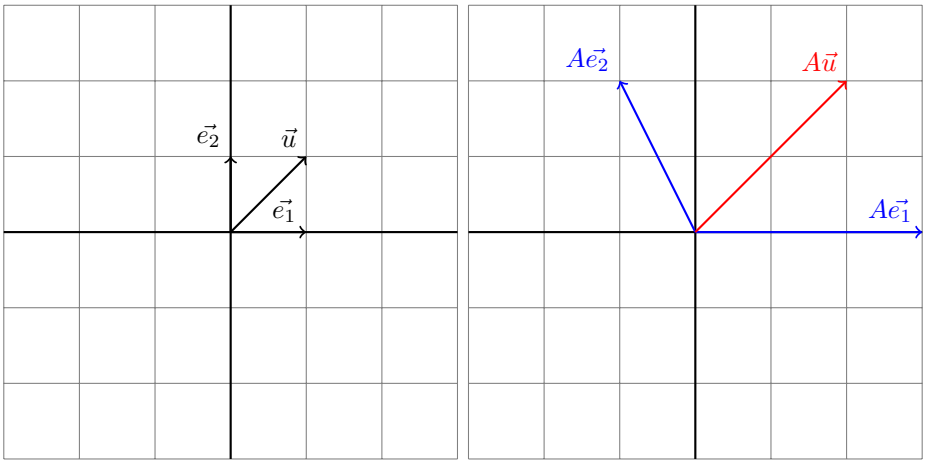
$$x_1 = x_2$$

We have that the eigenvector is any vector of the form $x_2 \begin{bmatrix} 1 \\ 1 \end{bmatrix}$



To set of all nontrivial solutions of $(A - \lambda I)\vec{x} = \vec{0}$ is the null space of $A - \lambda I$, called the **eigenspace**. $(A - \lambda I)\vec{x} = \vec{0}$ has nontrivial solutions only when $\det(A - \lambda I) = 0$.

Let's look at that example from a slightly different perspective. Let's take the eigenvector (1, 1) we found for A and the standard basis vectors $\vec{e_1}$ and $\vec{e_2}$ and watch how they all change when we apply the transformation A.



What about Rotations?

In 2D, there are no eigenvectors for a rotation transformation. There are no $\mathbb{R}^2$ vectors that you can rotate that remain as a scaled version of themselves. In $\mathbb{R}^3$ however, the eigenvector for a rotation will be the axis of rotation.