Lecture 11: MATH 342W: Introduction to Data Science and Machine Learning

Sergio E. Garcia Tapia*

March 13, 2025 (last updated March 14, 2025)

Gram-Schmidt Orthogonalization

Last time we began discussing the Gram-Schmidt Orthogonalization procedure. Suppose $X \in \mathbb{R}^{n\times m}$, where $m \leq n$ of full rank:

$$X = \begin{bmatrix} \uparrow & \uparrow & \cdots & \uparrow \\ \mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_m \\ \downarrow & \downarrow & \cdots & \downarrow \end{bmatrix}$$

where $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_m$ are column vectors in $\mathbb{R}^n$. Then the procedure produces an orthogonal matrix Q such that $colsp[X] = colsp[Q]$ as follows:

- **Step 1**: (Orthogonalize): Here, the technique involves removing the projection of the current vector under consideration onto the span of the previously constructed vectors, thereby yielding at each step a vector that is orthogonal to the previous ones:

$$\begin{aligned} \mathbf{v}_1 &:= \mathbf{x}_1 \\ \mathbf{v}_2 &:= \mathbf{x}_2 - \operatorname*{proj}_{\text{span}(\mathbf{v}_1)}(\mathbf{x}_2) \\ \mathbf{v}_3 &:= \mathbf{x}_3 - \operatorname*{proj}_{\text{span}(\mathbf{v}_1)}(\mathbf{x}_3) - \operatorname*{proj}_{\text{span}(\mathbf{v}_2)}(\mathbf{x}_3) \\ &\vdots := \vdots \\ \mathbf{v}_k &:= \mathbf{x}_k - \sum_{j=1}^{k-1} \operatorname*{proj}_{\text{span}(\mathbf{v}_j)}(\mathbf{x}_k), \quad 2 \leq k \leq m \end{aligned}$$

  Note that at each step, we have $\text{span}(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_k) = \text{span}(\mathbf{v}_1, \mathbf{v_2}, \ldots \mathbf{v}_k)$ for all $1 \leq k \leq m$.

- **Step 2**: (Normalize): The list $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_m$ is an *orthogonal list*, and here we divide each vector by its length to make them all length 1, yielding an *orthonormal list*:

$$\mathbf{q}_k := \frac{\mathbf{v}_k}{\|\mathbf{v}_k\|}, \quad 1 \leq k \leq m$$

*Based on lectures of Dr. Adam Kapelner at Queens College. See also the course GitHub page.

Hence, $(\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_m)$ is an orthonormal list, satisfying $\text{span}(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_k) = \text{span}(\mathbf{q}_1, \mathbf{q_2}, \ldots \mathbf{q}_k)$ for all $1 \leq k \leq m$, and

$$\mathbf{q}_i^\top \mathbf{q}_j = \begin{cases} 1 & \text{if } i \neq j \\ 0 & \text{otherwise.} \end{cases}$$

QR Decomposition

The Gram-Schmidt Orthogonalization procedure gives us a way to *factor* or *decompose* an full-rank $n \times m$ matrix X as follows:

$$X = QR$$

$$\underbrace{\begin{bmatrix} \uparrow & \uparrow & \cdots & \uparrow \\ \mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_m \\ \downarrow & \downarrow & \cdots & \downarrow \end{bmatrix}}_{n \times m} = \underbrace{\begin{bmatrix} \uparrow & \uparrow & \cdots & \uparrow \\ \mathbf{q}_1 & \mathbf{q}_2 & \cdots & \mathbf{q}_m \\ \downarrow & \downarrow & \cdots & \downarrow \end{bmatrix}}_{n \times m} \underbrace{\begin{bmatrix} \uparrow & \uparrow & \cdots & \uparrow \\ \mathbf{r}_1 & \mathbf{r}_2 & \cdots & \mathbf{r}_m \\ \downarrow & \downarrow & \cdots & \downarrow \end{bmatrix}}_{m \times m \text{ (square)}}$$

where

- $X \in \mathbb{R}^{n \times m}$ with $m \leq n$ and is full rank, in the original basis.
- $Q \in \mathbb{R}^{n \times m}$ with the same column space as X, but expressed with an orthonormal basis.
- $R \in \mathbb{R}^{m \times m}$ is a *change of basis matrix*, and it is upper-triangular of full rank.

Though not crucial for our course, let's go through the exercise of computing R. The kth column of R consists exactly of the coefficients r_{ik} needed to express $\mathbf{x}_k$, the kth column of X, as a linear combinations of the columns of Q. We can reverse-engineer the Gram-Schmidt method to determine what these coefficients should be. We start from Step 1 of Gram-Schmidt, where we re-arrange by isolating $\mathbf{x}_k$:

$$\begin{aligned} \mathbf{x}_1 &= \mathbf{v}_1 \\ \mathbf{x}_2 &= \mathbf{v}_2 + \underset{\text{span}(\mathbf{v}_1)}{\text{proj}}(\mathbf{x}_2) \\ \mathbf{x}_3 &= \mathbf{v}_3 + \underset{\text{span}(\mathbf{v}_1)}{\text{proj}}(\mathbf{x}_3) + \underset{\text{span}(\mathbf{v}_2)}{\text{proj}}(\mathbf{x}_3) \\ &\vdots = \vdots \\ \mathbf{x}_k &= \mathbf{v}_k + \sum_{j=1}^{k-1} \underset{\text{span}(\mathbf{v}_j)}{\text{proj}}(\mathbf{x}_k), \quad 2 \leq k \leq m \end{aligned}$$

The important thing to note is that $\mathbf{x}_k \in \text{span}(\mathbf{v}_1, \ldots, \mathbf{v}_k)$, and since the latter equals $\text{span}(\mathbf{q}_1, \ldots, \mathbf{q}_k)$, we have $\mathbf{x}_k \in \text{span}(\mathbf{q}_1, \ldots, \mathbf{q}_k)$, we can write

$$\mathbf{x}_k = \sum_{i=1}^{k} r_{ik} \mathbf{q}_i, \quad 1 \leq k \leq m$$

In particular, $\mathbf{q}_{k+1}, \ldots, \mathbf{q}_m$ are not in this sum, and so, $r_{ij} = 0$ if $i > j$, making R upper-triangular as we claimed. Here is one way to proceed. For example, the fact that $\mathbf{x}_1 = \mathbf{v}_1$ and $\mathbf{q}_1 := \mathbf{v}_1/\|\mathbf{v}_1\|$, we can write

$$\mathbf{x}_1 = \mathbf{v}_1 = \|\mathbf{v}_1\|\mathbf{q}_1 = \|\mathbf{x}_1\|\mathbf{q}_1$$

so that $r_{11} = 0$ and $r_{1j} = 0$ for $j > 1$. Next, for $\mathbf{x}_2$, we can write

$$\mathbf{x}_2 = b\mathbf{q}_1 + c\mathbf{q}_2$$

Since $\mathbf{q}_1$ and $\mathbf{q}_2$ are orthogonal, the projection of $\mathbf{x}_2$ onto $\text{span}(\mathbf{q}_1, \mathbf{q}_2)$ is precisely $H_1 + H_2$, where H_1 orthogonal projects onto $\text{span}(\mathbf{q}_1)$ and H_2 orthogonally projects onto $\text{span}(\mathbf{q}_2)$ by $\mathbf{q}_1$ and $\mathbf{q}_2$ (we proved this fact). Since $\mathbf{x}_2 \in \text{span}(\mathbf{q}_1, \mathbf{q}_2)$, it is unchanged by the projection, so we have

$$\begin{aligned}
b\mathbf{q}_1 + c\mathbf{q}_2 &= \mathbf{x}_2 \\
&= (H_1 + H_2)\mathbf{x}_2 && \text{(since } \mathbf{x}_2 \in \text{span}(\mathbf{q}_1, \mathbf{q}_2)\text{)} \\
&= H_1\mathbf{x}_2 + H_2\mathbf{x}_2 \\
&= (\mathbf{q}_1\mathbf{q}_1^\top)\mathbf{x}_2 + (\mathbf{q}_2\mathbf{q}_2^\top)\mathbf{x}_2 && \text{(definition of orthogonal projection)} \\
&= \mathbf{q}_1(\mathbf{q}_1^\top\mathbf{x}_2) + \mathbf{q}_2(\mathbf{q}_2^\top\mathbf{x}_2) && \text{(Associativity)} \\
&= (\mathbf{q}_1^\top\mathbf{x}_2)\mathbf{q}_1 + (\mathbf{q}_2^\top\mathbf{x}_2)\mathbf{q}_2
\end{aligned}$$

Hence, $r_{12} = b = \mathbf{q}_1^\top\mathbf{x}_2$ and $r_{22} = c = \mathbf{q}_2^\top\mathbf{x}_2$. We can certainly proceed this way. A different approach involves exploiting the orthonormality of the list $\mathbf{q}_1, \ldots, \mathbf{q}_k$ to compute the coefficients by multiplying by $\mathbf{q}_j^\top$ on the left:

$$\begin{aligned}
\mathbf{q}_j^\top\mathbf{x}_k &= \mathbf{q}_j^\top\left(\sum_{i=1}^{k} r_{ik}\mathbf{q}_i\right) \\
&= \sum_{i=1}^{k} r_{ik}\mathbf{q}_j^\top\mathbf{q}_i \\
&= r_{jk} && (\mathbf{q}_j^\top\mathbf{q}_i = 0 \text{ if } i \neq j)
\end{aligned}$$

Hence, we have

$$\mathbf{x}_k = \sum_{i=1}^{k}(\mathbf{q}_i^\top\mathbf{x}_k)\mathbf{q}_i$$

and we can express R as

$$R = \begin{bmatrix} \mathbf{q}_1^\top\mathbf{x}_1 & \mathbf{q}_1^\top\mathbf{x}_2 & \cdots & \cdots & \mathbf{q}_1^\top\mathbf{x}_m \\ 0 & \mathbf{q}_2^\top\mathbf{x}_2 & \cdots & \cdots & \mathbf{q}_2^\top\mathbf{x}_m \\ 0 & 0 & \ddots & \cdots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & \mathbf{q}_m^\top\mathbf{x}_m \end{bmatrix} \iff r_{ij} = \begin{cases} \mathbf{q}_i^\top\mathbf{x}_j & \text{if } i \leq j \\ 0 & \text{if } i > j \end{cases}$$

Computing the Orthogonal Projection Matrix

Here is a typical test question. Recall we have shown that orthogonal matrices are unique, and it is given by

$$H = X(X^\top X)^{-1}X^\top$$

If we decompose $X \in \mathbb{R}^{n\times m}$ into QR (again we assume X is full rank with $m \leq n$), we can show that $H = QQ^\top$. Recall R is square and full rank, so it is invertible:

$$\begin{aligned}
H &= (QR)((QR)^\top(QR))^{-1}(QR)^\top \\
&= QR(R^\top \underbrace{Q^\top Q}_{I_m} R)^{-1}(QR)^\top && ((AB)^\top = B^\top A^\top) \\
&= QR(R^\top R)^{-1}R^\top Q^\top && ((AB)^\top = B^\top A^\top) \\
&= Q\underbrace{RR^{-1}}_{I_m}\underbrace{(R^\top)^{-1}R^\top}_{I_m}Q^\top && ((AB)^{-1} = B^{-1}A^{-1}) \\
&= QQ^\top
\end{aligned}$$

The Monotonicity of SSR

The Gram-Schmidt Orthogonalization procedure and QR decomposition are techniques described in the context of a liner algebra course. Why did we bother to discuss them at length? What do we gain from the change of basis into an orthonormal basis? In this section, we use the orthonormal basis to discuss error metrics. Let X be our design matrix with n rows, $p+1$ columns, first column being $\mathbf{1}_n$, and full rank. In an earlier lecture, we proved that

$$SST = SSR + SSE$$
$$\sum_{i=1}^{n}(y_i - \bar{y})^2 = \sum_{i=1}^{n}(\hat{y}_i - \bar{y})^2 + \sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

Suppose we decide to include another feature (increase p), but do not add any new observations (n remains fixed). Then the SST stays fixed since the y_i remain unchanged, and so does the average $\bar{y}$. We want to investigate whether SSR, SSE, and our other errors metrics change as a result of this increase in p. We can focus solely on SSR, because the equation above implies that

$$SSR \uparrow \iff SSE \downarrow \iff R^2 \uparrow$$

This motivates the following computation:

$$\begin{aligned}
SSR &= \sum_{i=1}^{n}(\hat{y}_i - \bar{y})^2 \\
&= (\hat{\mathbf{y}}_i - \bar{y}\mathbf{1}_n)^\top(\hat{\mathbf{y}}_i - \bar{y}\mathbf{1}_n) \\
&= \hat{\mathbf{y}}_i^\top\hat{\mathbf{y}}_i - \hat{\mathbf{y}}_i^\top\bar{y}\mathbf{1}_n - \bar{y}\mathbf{1}_n^\top\hat{\mathbf{y}}_i + \bar{y}\mathbf{1}_n^\top\bar{y}\mathbf{1}_n \\
&= \|\hat{\mathbf{y}}_i\|^2 - 2\bar{y}\hat{\mathbf{y}}_i^\top\mathbf{1}_n + (\bar{y})^2\mathbf{1}_n^\top\mathbf{1}_n \qquad (1)
\end{aligned}$$

where the last line uses the fact that $\hat{\mathbf{y}}_i^\top \mathbf{1}_n = \mathbf{1}_n^\top \hat{\mathbf{y}}_i$. First, note

$$\mathbf{1}_n^\top \mathbf{1}_n = \begin{bmatrix} 1 & 1 \cdots & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} = n$$

Next, recall that $\hat{\mathbf{y}} = H\mathbf{y}$, where H is the orthogonal projection matrix onto $col sp[X]$. Thus,

$$\begin{aligned}
\hat{\mathbf{y}}^\top \mathbf{1}_n &= (H\mathbf{y})^\top \mathbf{1}_n \\
&= \mathbf{y}^\top H^\top \mathbf{1}_n && ((AB)^\top = B^\top A^\top) \\
&= \mathbf{y}^\top H \mathbf{1}_n && (H \text{ is symmetric}) \\
&= \mathbf{y}^\top \mathbf{1}_n && (\mathbf{1}_n \in col sp[X]) \\
&= \begin{bmatrix} y_1 & y_2 & \cdots & y_n \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \\
&= n\bar{y}
\end{aligned}$$

Thus, Equation 1 becomes:

$$\begin{aligned}
SSR &= \|\hat{\mathbf{y}}_i\|^2 - 2\bar{y} \cdot n(\bar{y}) + n(\bar{y})^2 \\
&= \|\hat{\mathbf{y}}_i\|^2 - n(\bar{y})^2 \qquad (2)
\end{aligned}$$

Next, we use QR be the decomposition of X, and make use of the orthonormal basis formed by the columns of Q to expand $\|\hat{\mathbf{y}}_i\|^2$ and further simplify Equation 2

$$\hat{\mathbf{y}} = \underset{col sp[X]}{\text{proj}}(\mathbf{y}) = QQ^\top \mathbf{y} = \sum_{j=0}^{p} \underset{\text{span}(\mathbf{q}_j)}{\text{proj}}(\mathbf{y})$$

Now, since the list $\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_p$, we can use the generalized Pythagorean Theorem:

$$\begin{aligned}
\|\hat{\mathbf{y}}\|^2 &= \left\| \sum_{j=0}^{p} \underset{\text{span}(\mathbf{q}_j)}{\text{proj}}(\mathbf{y}) \right\|^2 \\
&= \sum_{j=0}^{p} \left\| \underset{\text{span}(\mathbf{q}_j)}{\text{proj}}(\mathbf{y}) \right\|^2 && (\text{by orthogonality}) \\
&= \left\| \underset{\text{span}(\mathbf{q}_0)}{\text{proj}}(\mathbf{y}) \right\|^2 + \sum_{j=1}^{p} \left\| \underset{\text{span}(\mathbf{q}_j)}{\text{proj}}(\mathbf{y}) \right\|^2
\end{aligned}$$

Note that from Gram-Schmidt, we saw that $\mathbf{x}_0 = \|\mathbf{x_0}\|\mathbf{q}_0$, and also by design we know that $\mathbf{x}_0 = \mathbf{1}_n$, so $\mathbf{q}_0 = \frac{1}{\sqrt{n}}\mathbf{1}_n$. Hence,

$$\begin{aligned}
\left\| \underset{\text{span}(\mathbf{q}_0)}{\text{proj}}(\mathbf{y}) \right\|^2 &= \left\| (\mathbf{q}_0^\top \mathbf{y})\mathbf{q}_0 \right\|^2 \\
&= |\mathbf{q}_0^\top \mathbf{y}|^2 \cdot \|\mathbf{q}_0\|^2 \\
&= \left| \frac{1}{\sqrt{n}} \mathbf{1}_n^\top \mathbf{y}_n \right|^2 \cdot 1^2 && (\|\mathbf{q}_0\| = 1) \\
&= \frac{1}{n} \cdot (\sum_{i=1}^{n} y_i)^2 \\
&= n(\bar{y})^2
\end{aligned}$$

Hence,

$$\|\hat{\mathbf{y}}\|^2 = n(\bar{y})^2 + \sum_{j=1}^{p} \left\| \underset{\text{span}(\mathbf{q}_j)}{\text{proj}}(\mathbf{y}) \right\|^2 \tag{3}$$

Finally, substituting into Equation 2:

$$\begin{aligned}
SSR &= \|\hat{\mathbf{y}}_i\|^2 - n(\bar{y})^2 \\
&= n(\bar{y})^2 + \sum_{j=1}^{p} \left\| \underset{\text{span}(\mathbf{q}_j)}{\text{proj}}(\mathbf{y}) \right\|^2 - n(\bar{y})^2 \\
&= \sum_{j=1}^{p} \left\| \underset{\text{span}(\mathbf{q}_j)}{\text{proj}}(\mathbf{y}) \right\|^2
\end{aligned}$$

We have arrived at our desired equation for SSR:

$$SSR = \sum_{j=1}^{p} \left\| \underset{\text{span}(\mathbf{q}_j)}{\text{proj}}(\mathbf{y}) \right\|^2 \tag{4}$$

This equation says that the SSR, which is a measurement of the fit, can be expressed as the sum of the squares of the lengths of the projections. More importantly, *as you add new features, the SSR increases*. To see this, suppose your friend walks in and claims to have an "amazing new predictor", $\mathbf{x}_{new}$, linearly independent of the columns of X, and claims that you can use it to predict $\hat{\mathbf{y}}$ much better. We add the new feature by binding a new column to our matrix:

$$X_{new} = \left[X \mid \mathbf{x}_{new} \right]$$

That is, we now have $p + 2$ columns in X. Say $X = QR$ is the QR decomposition of X. Note that since $col\ell sp[X] = col\ell sp[Q]$, this is equivalent to modifying Q as follows:

$$Q_{new} = \left[Q \mid \mathbf{q}_{new} \right]$$

According to Equation 4, our SSR changes as follows:

$$\begin{aligned} SSR_{new} &= \sum_{j=1}^{p+2} \left\| \underset{\text{span}(\mathbf{q}_j)}{\text{proj}}(\mathbf{y}) \right\|^2 \\ &= \left(\sum_{j=1}^{p+1} \left\| \underset{\text{span}(\mathbf{q}_j)}{\text{proj}}(\mathbf{y}) \right\|^2 \right) + \left\| \underset{\text{span}(\mathbf{q}_{new})}{\text{proj}}(\mathbf{y}) \right\|^2 \\ &= SSR + \left\| \underset{\text{span}(\mathbf{q}_{new})}{\text{proj}}(\mathbf{y}) \right\|^2 \end{aligned}$$

The quantity $\left\| \underset{\text{span}(\mathbf{q}_{new})}{\text{proj}}(\mathbf{y}) \right\|^2$ is non-negative, so

$$SSR_{new} \geq SSR \iff SSE_{new} \leq SSE \iff R^2_{new} \geq R^2$$

In fact, $\left\| \underset{\text{span}(\mathbf{q}_{new})}{\text{proj}}(\mathbf{y}) \right\|^2 = 0$ if and only if $\mathbf{q}_{new} \perp \mathbf{y}$, which is nearly impossible in the real world. Thus, we can often think of of SSR as being strictly monotonically increasing with respect to the number of features.

Overfitting

Consider your friend who earlier suggested a new amazing feature $\mathbf{x}_{new}$. They return and say "I made up $\mathbf{x}_{new}$ with random numbers" (the audacity). Mathematically, our predictions are better – we have a "better fit", but realistically this cannot be. Think back to lecture 1; the x's are supposed to be proxies to the z's, and since these random numbers have nothing to do with the z's, we are fitting noise (i.e. we are fitting δ), which is impossible. The new model will predict worse in the future, and hence generalization error increases. Sad to say, but the error metrics we have used, the SSR, SSE, and R^2 have been a total lie; they are **dishonest error metrics**.

Indeed, we can manipulate the SSE to be as close to 0 as you wish, and R^2 to be as close to 1 as you wish, simply by adding more random junk like our supposed friend. What about the other error metrics?

$$MSE := \frac{SSE}{n-(p+1)}, \quad RMSE := \sqrt{\frac{SSE}{n-(p+1)}}$$

These metrics are also lies, but they have some "insurance". If p increases, then $n-(p+1)$ decreases, and hence, MSE and $RMSE$ increase. However, they are still not recommended. What happens if we keep going? Suppose that $\mathbf{x}_{rand} \in \mathbb{R}^n$ is a random vector. It's unlikely that $\mathbf{x}_{rand}$ will be orthogonal to $\mathbf{y}$, so by projecting $\mathbf{y}$ onto $\text{span}(\mathbf{x}_{rand})$, we get a little more of $\hat{\mathbf{y}}$ (see Figure 1). This is called **chance capitalization**. You think it's a fit, but it's not, and you are tricking yourself into thinking you got somewhere, but you did not. This phenomenon is called **overfitting**.

Let's continue to add more random vectors to X until it becomes $n \times n$, and assume it is full rank. Then X is invertible, and the orthogonal projection matrix is now given

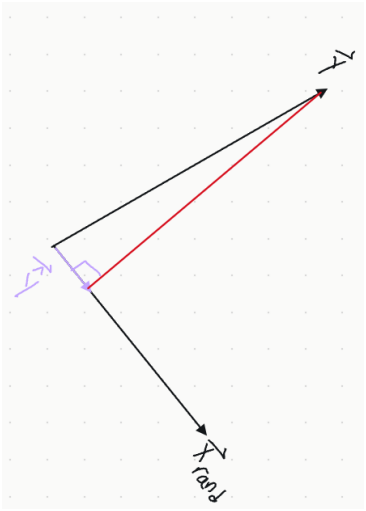

Figure 1: Projectiong $\mathbf{y}$ onto a random vector, resulting in chance capitalization

by

$$\begin{aligned} H &= X(X^\top X)^{-1}X^\top \\ &= \underbrace{XX^{-1}}_{I_n}\underbrace{(X^\top)^{-1}X^\top}_{I_n} \\ &= I_n \end{aligned}$$

Now if we project $\mathbf{y}$ onto the column space of X, we get

$$\begin{aligned} \hat{\mathbf{y}} &= H\mathbf{y} = I_n\mathbf{y} = \mathbf{y} \\ \mathbf{e} &= \mathbf{y} - \hat{\mathbf{y}} = \mathbf{0}_n \end{aligned}$$

Hence, $SSE = \|\mathbf{e}\|^2 = 0$, and $R^2 = 1 - \frac{SSE}{SST} = 1$; we have a perfect fit. This is clearly a problem because we can take a computer, fill our design matrix with random junk, and we get a perfect fit. Let's visualize overfitting in the case where $p = 1$. Then we have $X \in \mathbb{R}^{n\times 2}$:

$$X = \begin{bmatrix} \mathbf{1}_n \mid \mathbf{x} \end{bmatrix}$$

With high confidence, we can say that our feature x is not the true driver z. However, if $n = 2$ (so we have two data points), we get a perfect fit. A scatterplot would look like Figure 2.

Honest Performance Metrics

We need honest performance metrics. These should approximate how well we predict in the future. Consider that our data set $\mathbf{D}$ is data we have collected in the past. Suppose we were omniscient and had $\mathbb{D}^*$, a data set of future data (see Figure 3). We could predict on X_* to get $\hat{\mathbf{y}}_*$, and consider $\mathbf{e}_* = \mathbf{y}_* - \hat{\mathbf{y}}_*$, with associated R^2_* and SSE_*. We will continue this discussion next time.

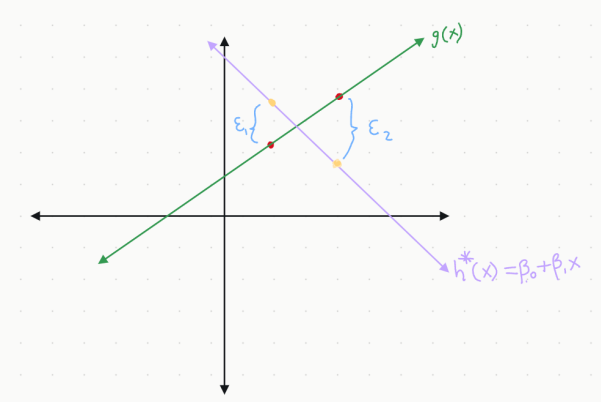

Figure 2: Overfitting with $n = 2$ and $p = 1$.

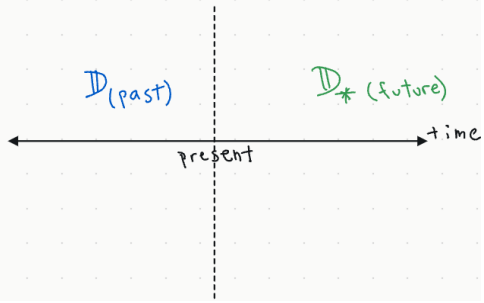

Figure 3: Leveraging future data to design honest error metrics.