Linear Regression as Prediction

36-462/36-662, Spring 2022

20 January 2022

Context

- We want to find patterns in our data which will let us predict one variable from another (or from more than one variable)
- The oldest and most widely-used prediction method is linear regression
- We're going to review linear regression, as a prediction method
- Without a lot of the mythology from 401
- But with an eye to more powerful methods

Optimal prediction in general

- We want to predict a *numerical* random variable Y
- We want a *one-number* ("point") prediction, not a range or a distribution
- We say how good our guess is by expected squared error

Optimal prediction in general (cont'd.)

What's the best *constant* guess for a random variable Y?

$$\begin{aligned}
\mu &= \operatorname*{argmin}_{m \in \mathbb{R}} \mathbb{E}\left[(Y-m)^2\right] && (1)\\
&= \operatorname*{argmin}_{m} \mathrm{Var}\left[(Y-m)\right] + \left(\mathbb{E}\left[Y-m\right]\right)^2 && (2)\\
&= \operatorname*{argmin}_{m} \mathrm{Var}\left[Y\right] + \left(\mathbb{E}\left[Y\right]-m\right)^2 && (3)\\
&= \operatorname*{argmin}_{m} \left(\mathbb{E}\left[Y\right]-m\right)^2 && (4)\\
&= \mathbb{E}\left[Y\right] && (5)
\end{aligned}$$

(Because: $\mathbb{E}\left[Z^2\right] = \mathrm{Var}\left[Z\right] + \left(\mathbb{E}\left[Z\right]\right)^2$, always)

Optimal prediction in general (cont'd.)

What's the best *function* of X to guess for Y?

$$\begin{aligned}
\mu &= \operatorname*{argmin}_{m:\mathcal{X}\mapsto\mathbb{R}} \mathbb{E}\left[(Y-m(X))^2\right] && (6)\\
&= \operatorname*{argmin}_{m} \mathbb{E}\left[\mathbb{E}\left[(Y-m(X))^2|X\right]\right] && (7)
\end{aligned}$$

For each x, best $m(x)$ is $\mathbb{E}[Y|X=x]$ (by previous slide)

$$\mu(x) = \mathbb{E}[Y|X=x]$$

Optimal prediction in general (cont'd.)

Learning arbitrary functions is hard!

Who knows what the right function might be?

What if we *decide* to make our predictions linear?

Optimal linear prediction with univariate predictor

Our prediction will be of the form

$$m(x) = a + bx$$

and we want the best a, b

Optimal linear prediction, univariate case

$$(\alpha, \beta) = \operatorname*{argmin}_{a,b} \mathbb{E}\left[(Y - (a+bX))^2\right]$$

Expand out that expectation, then take derivatives and set them to 0

The intercept

$$\begin{aligned}
\mathbb{E}\left[(Y-(a+bX))^2\right] &= \mathbb{E}\left[Y^2\right] - 2\mathbb{E}\left[Y(a+bX)\right] + \mathbb{E}\left[(a+bX)^2\right] && (8)\\
&= \mathbb{E}\left[Y^2\right] - 2a\mathbb{E}\left[Y\right] - 2b\mathbb{E}\left[YX\right] + && (9)\\
&\quad a^2 + 2ab\mathbb{E}\left[X\right] + b^2\mathbb{E}\left[X^2\right] && (10)\\
\frac{\partial}{\partial a}\mathbb{E}\left[(Y-(a+bX))^2\right]\Big|_{a=\alpha,b=\beta} &= -2\mathbb{E}\left[Y\right] + 2\alpha + 2\beta\mathbb{E}\left[X\right] = 0 && (11)\\
\alpha &= \mathbb{E}\left[Y\right] - \beta\mathbb{E}\left[X\right] && (12)
\end{aligned}$$

$\therefore$ optimal linear predictor $m(X) = \alpha + \beta X$ looks like

$$\begin{aligned}
m(X) &= \alpha + \beta X && (13)\\
&= \mathbb{E}\left[Y\right] - \beta\mathbb{E}\left[X\right] + \beta X && (14)\\
&= \mathbb{E}\left[Y\right] + \beta(X - \mathbb{E}\left[X\right]) && (15)
\end{aligned}$$

The optimal linear predictor only cares about how far X is from its expectation $\mathbb{E}[X]$ And when $X = \mathbb{E}[X]$, we will always predict $\mathbb{E}[Y]$

The slope

$$\begin{aligned}
\frac{\partial}{\partial b}\mathbb{E}\left[(Y-(a+bX))^2\right]\Big|_{a=\alpha,b=\beta} &= -2\mathbb{E}\left[YX\right]+2\alpha\mathbb{E}\left[X\right]+2\beta\mathbb{E}\left[X^2\right]=0 && (16)\\
0 &= -\mathbb{E}\left[YX\right]+(\mathbb{E}\left[Y\right]-\beta\mathbb{E}\left[X\right])\mathbb{E}\left[X\right]+\beta\mathbb{E}\left[X^2\right] && (17)\\
0 &= \mathbb{E}\left[Y\right]\mathbb{E}\left[X\right]-\mathbb{E}\left[YX\right]+\beta(\mathbb{E}\left[X^2\right]-\mathbb{E}\left[X\right]^2) && (18)\\
0 &= -\mathrm{Cov}\left[Y,X\right]+\beta\mathrm{Var}\left[X\right] && (19)\\
\beta &= \frac{\mathrm{Cov}\left[Y,X\right]}{\mathrm{Var}\left[X\right]} && (20)
\end{aligned}$$

- If we replace X with $X' = X - \mathbb{E}\left[X\right]$, β doesn't change
- If we replace Y with $Y' = Y - \mathbb{E}\left[Y\right]$, β doesn't change
- $\therefore$ centering the variables doesn't change the slope

The optimal linear predictor of Y from X

The optimal linear predictor of Y from a single X is *always*

$$\alpha+\beta X = \mathbb{E}\left[Y\right]+\left(\frac{\mathrm{Cov}\left[X,Y\right]}{\mathrm{Var}\left[X\right]}\right)(X-\mathbb{E}\left[X\right])$$

What did we not assume?

- That the true relationship between Y and X is linear
- That anything is Gaussian
- That anything has constant variance
- That anything is independent or even uncorrelated

  NONE OF THAT MATTERS for the optimal linear predictor

The prediction errors average out to zero

$$\begin{aligned}
\mathbb{E}\left[Y-m(X)\right] &= \mathbb{E}\left[Y-(\mathbb{E}\left[Y\right]+\beta(X-\mathbb{E}\left[X\right]))\right] && (21)\\
&= \mathbb{E}\left[Y\right]-\mathbb{E}\left[Y\right]-\beta(\mathbb{E}\left[X\right]-\mathbb{E}\left[X\right])=0 && (22)
\end{aligned}$$

- If they didn't average to zero, we'd adjust the coefficients until they did
- **Important**: In general, $\mathbb{E}\left[Y-m(X)|X\right] \neq 0$

The prediction errors are uncorrelated with X

$$\begin{aligned}
\mathrm{Cov}\left[X, Y - m(X)\right] &= \mathbb{E}\left[X(Y - m(X))\right] \text{ (by previous slide)} && (23)\\
&= \mathbb{E}\left[X(Y - \mathbb{E}\left[Y\right] - \frac{\mathrm{Cov}\left[Y,X\right]}{\mathrm{Var}\left[X\right]}(X - \mathbb{E}\left[X\right]))\right] && (24)\\
&= \mathbb{E}\left[XY - X\mathbb{E}\left[Y\right] - \frac{\mathrm{Cov}\left[Y,X\right]}{\mathrm{Var}\left[X\right]}(X^2) + \frac{\mathrm{Cov}\left[Y,X\right]}{\mathrm{Var}\left[X\right]}(X\mathbb{E}\left[X\right])\right] && (25)\\
&= \mathbb{E}\left[XY\right] - \mathbb{E}\left[X\right]\mathbb{E}\left[Y\right] - \frac{\mathrm{Cov}\left[Y,X\right]}{\mathrm{Var}\left[X\right]}\mathbb{E}\left[X^2\right] + \frac{\mathrm{Cov}\left[Y,X\right]}{\mathrm{Var}\left[X\right]}(\mathbb{E}\left[X\right])^2 && (26)\\
&= \mathrm{Cov}\left[X,Y\right] - \frac{\mathrm{Cov}\left[Y,X\right]}{\mathrm{Var}\left[X\right]}(\mathrm{Var}\left[X\right]) && (27)\\
&= 0 && (28)
\end{aligned}$$

- If the errors weren't uncorrelated with X, we'd adjust the coefficients until they were

The prediction errors are uncorrelated with X

Alternate take:

$$\begin{aligned}
\mathrm{Cov}\left[X, Y - m(X)\right] &= \mathrm{Cov}\left[X,Y\right] - \mathrm{Cov}\left[X, \alpha + \beta X\right] && (29)\\
&= \mathrm{Cov}\left[Y,X\right] - \mathrm{Cov}\left[X, \beta X\right] && (30)\\
&= \mathrm{Cov}\left[Y,X\right] - \beta\mathrm{Cov}\left[X,X\right] && (31)\\
&= \mathrm{Cov}\left[Y,X\right] - \beta\mathrm{Var}\left[X\right] && (32)\\
&= \mathrm{Cov}\left[Y,X\right] - \mathrm{Cov}\left[Y,X\right] = 0 && (33)
\end{aligned}$$

How big are the prediction errors?

$$\begin{aligned}
\mathrm{Var}\left[Y - m(X)\right] &= \mathrm{Var}\left[Y - \alpha - \beta X\right] && (34)\\
&= \mathrm{Var}\left[Y - \beta X\right] && (35)\\
& && (36)
\end{aligned}$$

> **After-class exercise**: Reduce this to an expression involving only $\mathrm{Var}\left[Y\right]$, $\mathrm{Var}\left[X\right]$ and $\mathrm{Cov}\left[Y,X\right]$; if you get the right answer you should see that it's $< \mathrm{Var}\left[Y\right]$ unless $\mathrm{Cov}\left[Y,X\right] = 0$

$\Rightarrow$ Optimal linear predictor is almost always better than nothing...

Multivariate case

- We try to predict Y from a whole bunch of variables
- Bundle those predictor variables into $\vec{X}$
- We need a vector of slope coefficients $\vec{\beta}$
- Solution:

$$m(\vec{X}) = \alpha + \vec{\beta}\cdot\vec{X} = \mathbb{E}\left[Y\right] + \mathrm{Var}\left[\vec{X}\right]^{-1}\mathrm{Cov}\left[\vec{X}, Y\right](\vec{X} - \mathbb{E}\left[\vec{X}\right])$$

and

$$\text{Var}\left[Y - m(\vec{X})\right] = \text{Var}\left[Y\right] - \text{Cov}\left[Y, \vec{X}\right]^T \text{Var}\left[\vec{X}\right]^{-1} \text{Cov}\left[Y, \vec{X}\right]$$

(Gory details in the back-up slides)

What we don't assume, again

- That the linear predictor is correct
- That anything is Gaussian
- Anything about the distributions of Y or $\vec{X}$

Estimation: Data, not the full distribution

- $\text{Var}\left[X\right]$, $\text{Cov}\left[Y, X\right]$, $\mathbb{E}\left[(Y - (a + bX))^2\right]$ all involve the *true* distribution, which we don't know
- $\therefore$ We can't just *calculate* β
- What we do know is $(x_1, y_1), (x_2, y_2), \ldots (x_n, y_n)$
  - That is, n pairs of predictor and outcome

Estimation: Ordinary Least Squares (OLS)

Set up the **mean squared error**, and minimize it:

$$MSE(a, b) = \frac{1}{n}\sum_{i=1}^{n}(y_i - (a + bx_i))^2 \tag{37}$$

$$(\hat{\alpha}, \hat{\beta}) \equiv \underset{a,b}{\text{argmin}}\, MSE(a, b) \tag{38}$$

Do the calculus:

$$\frac{\partial MSE}{\partial a} = \frac{1}{n}\sum_{i=1}^{n} -2(y_i - (a + bx_i)) \tag{39}$$

$$\frac{\partial MSE}{\partial b} = \frac{1}{n}\sum_{i=1}^{n} -2(y_i - (a + bx_i))x_i \tag{40}$$

$$\hat{\alpha} = \overline{y} - \hat{\beta}\overline{x} \tag{41}$$

$$\hat{\beta} = \frac{\overline{yx} - \overline{y}\,\overline{x}}{\overline{x^2} - \overline{x}^2} = \frac{\widehat{\text{Cov}\left[Y, X\right]}}{\widehat{\text{Var}\left[X\right]}} \tag{42}$$

(with $\overline{x}$ = sample mean of x, etc.)

Optimum vs. estimate (I)

- Optimal linear model *versus* Linear model estimated by OLS:

$$\alpha + \beta x = \mathbb{E}\left[Y\right] + \left(\frac{\text{Cov}\left[X, Y\right]}{\text{Var}\left[X\right]}\right)(x - \mathbb{E}\left[X\right]) \tag{43}$$

$$\hat{\alpha} + \hat{\beta} x = \overline{y} + \frac{\widehat{\text{Cov}\left[Y, X\right]}}{\widehat{\text{Var}\left[X\right]}}(x - \overline{x}) \tag{44}$$

- These are not the same, but we can hope they're close, and increasingly close with more data

When does OLS/plug-in work?

- "Work" = converge on the optimal linear predictor
- Jointly sufficient conditions:

1. Sample means converge on expectation values
2. Sample covariances converge on true covariance
3. Sample variances converge on true, invertible variance

- Then by continuity OLS coefficients converge on true β
- None of this requires that the linear model is right, that anything is Gaussian, etc.

Optimum vs. estimate (II)

- A little more subtle: for every *fixed* (a, b), by the law of large numbers,

$$MSE(a, b) \to \mathbb{E}\left[(Y - (a + bX))^2\right]$$

- Mean squared error $\to$ expected squared error
- This is *almost* enough to ensure that minimize MSE will converge on the true optimum
- We'll come back to this, because this approach generalizes better to other models and loss functions

What do the estimates look like?

- Bundle all the regressor values into an $n \times (p+1)$ matrix $\mathbf{x}$, with an extra column of 1s
- Bundle all the regressand values into an $n \times 1$ matrix $\mathbf{y}$
- Then the $(p+1) \times 1$ matrix of least-squares coefficients $\hat{\beta}$ is

$$\hat{\beta} = (\mathbf{x}^T\mathbf{x})^{-1}\mathbf{x}^T\mathbf{y}$$

(as you learned in linear models)

- This is really

$$\hat{\beta} = \left(\frac{1}{n}\mathbf{x}^T\mathbf{x}\right)^{-1}\left(\frac{1}{n}\mathbf{x}^T\mathbf{y}\right) \quad (45)$$

$$= (\text{sample variance matrix of regressors})^{-1}(\text{sample covariances between regressors and response}) \quad (46)$$

What do the predictions look like?

- The prediction at an arbitrary point $\vec{x} = (x_1, x_2, \ldots x_p)$ is

$$\hat{m}(\vec{x}) = \begin{bmatrix}1 & x_1 & x_2 & \ldots & x_p\end{bmatrix} \hat{\beta} = \begin{bmatrix}1 & x_1 & x_2 & \ldots & x_p\end{bmatrix} \left((\mathbf{x}^T\mathbf{x})^{-1}\mathbf{x}^T\mathbf{y}\right)$$

- The $n \times 1$ matrix of fitted values, at the training points, is

$$\hat{\mathbf{m}} = \mathbf{x}(\mathbf{x}^T\mathbf{x})^{-1}\mathbf{x}^T\mathbf{y}$$

- The matrix $\mathbf{h} \equiv \mathbf{x}(\mathbf{x}^T\mathbf{x})^{-1}\mathbf{x}^T$ is the **weight matrix**, **influence matrix** or **hat matrix** and says how much y_j matters to the prediction of y_i

Fitted values and other predictions are weighted sums of the observations

$$\hat{m}(\vec{x}) = \vec{x}\hat{\beta} = \vec{x}(\mathbf{x}^T\mathbf{x})^{-1}\mathbf{x}^T\mathbf{y} \quad (47)$$
$$\hat{\mathbf{m}} = \mathbf{x}(\mathbf{x}^T\mathbf{x})^{-1}\mathbf{x}^T\mathbf{y} \quad (48)$$

- Every prediction we make is linear in $\mathbf{y}$, it's a weighted sum of all the observed values of the response
- How much weight do we put on y_i when we're predicting at $\vec{x}$?

Explicit form of the weights for OLS

- For univariate OLS, if we're trying to predict at the point x and built the model with data $(x_1, y_1), \ldots (x_n, y_n)$, we end up with:

$$\hat{m}(x) = \overline{y} + \hat{\beta}_1(x - \overline{x}) \quad (49)$$
$$= \sum_{i=1}^{n} \frac{1}{n}\left(1 + \frac{(x-\overline{x})(x_i - \overline{x})}{\hat{\sigma}_x^2}\right) y_i \quad (50)$$

  - We give more weight to y_i if x_i is far from $\overline{x}$
  - and we give more weight to y_i if x is far from $\overline{x}$
  - Weights don't care about $x - x_i$, which is stupid, but also required to get a straight line
  - Same idea, with more algebra, for multiple regressors

Generalizing: linear smoothers

- Linear regression is a special case of a **linear smoother**

$$\widehat{\mu}(\vec{x}) = \sum_{i=1}^{n} w(\vec{x}, \vec{x}_i) y_i$$

- Notice: Linear in the y's, not in $\vec{x}$
- General idea: predict at $\vec{x}$ by finding similar $\vec{x}_i$'s and averaging their y_i's
- Different choices of w are different ideas about "similar" and about weighted averaging
- Most prediction methods used in practice are linear smoothers
  - Nearest neighbors
  - Trees and forests
  - Kernels
  - Neural networks, a.k.a. "deep learning", a.k.a. "artificial intelligence"

What about the rest of your linear models course?

- Say you want to test whether $\beta_{37} = 1$, or to give a confidence set for (α, β_{12})
- That's hard to do *explicitly* without more assumptions
- Generally, you need to know the **sampling distribution** of $\hat{\beta}$

What about the rest of your linear models course? (cont'd)

- The usual ("401") assumptions give nice formulas for the sampling distribution of parameter estimators, and so for parameter inference:

1. The true regression function is exactly linear.

2. $Y = \alpha + \vec{X} \cdot \vec{\beta} + \epsilon$ where ϵ is *independent* of x.
3. ϵ is independent across observations.
4. $\epsilon \sim \mathcal{N}(0, \sigma^2)$.

- Assuming (1)–(4), the usual formulas (embedded in R) are exactly right
  - Gaussian distribution for $\hat{\beta}$, centered at true β
- Assuming (1)–(3), non-Gaussian noise, and the influence of any one observation $\to 0$, then the usual formulas hold as $n \to \infty$
  - Get back the Gaussian distribution by a (complicated) central limit theorem
- **Do not use those formulas without checking those assumptions**

The most important assumption to check

- If the **true** regression function is *exactly* linear, then

$$\mathbb{E}\left[Y - (\alpha + \vec{X}\beta)|\vec{X} = \vec{x}\right] = 0$$

  for all $\vec{x}$
- So

$$\mathbb{E}\left[Y - (\hat{\alpha} + \vec{X}\hat{\beta})|\vec{X} = \vec{x}\right] \approx 0$$

- **The residuals should have conditional mean 0 everywhere** if the linear model is right
  - This is something we can check using nonlinear regression methods

Summing up

- We can always *decide* to use a linear predictor, $m(\vec{X}) = \alpha + \vec{\beta} \cdot \vec{X}$
- The optimal linear predictor of Y from $\vec{X}$ always takes the same form:

$$m(Y) = \mathbb{E}[Y] + \text{Var}\left[\vec{X}\right]^{-1} \text{Cov}\left[Y, \vec{X}\right] (\vec{X} - \mathbb{E}\left[\vec{X}\right])$$

- Doing linear prediction requires finding those covariances
- We usually estimate those covariances (implicitly) by ordinary least squares
- OLS converges pretty robustly to the optimal linear predictor (not to the truth)
- The usual "401" assumptions are needed for *parameter* inference, not point prediction
- Once we use OLS, all our predictions are weighted sums of the observations
- The weights for linear regression are weird and implausible
- We're going to explore other ways of weighting
- We're also going to explore minimize other measures of prediction error, over other classes of predictors

A final thought

When you're fundraising, it's AI
When you're hiring, it's ML
When you're implementing, it's linear regression

- Baron Schwartz, 15 November 2017[1]

[1]Since-deleted tweet, but see e.g. [https://twitter.com/bc238dev/status/1225150435729666048]. I've often seen the last line quoted as "it's logistic regression", which fits with computer science's emphasis on classification rather than regression, but so far as I can work out that's a later mutation.

Backup: Further reading

- See Berk (2008), chapter 1, and Hastie, Tibshirani, and Friedman (2009), sections 2.3.1 and 2.6, and chapter 3 (especially through section 3.4), for more on linear regression as a prediction method
- If you want more from the point of view taken here, see Shalizi (n.d.), chapter 2; if you want a *lot* more, see Shalizi (2015)
- If you need to do inference about the coefficients but the usual assumptions don't hold, your best bet is the bootstrap (covered in 402, and in some excellent textbooks like Davison and Hinkley (1997)); some of the complicated procedures developed in econometrics for "robust standard errors" in linear models turn out to be equivalent to simple bootstraps (Buja et al. 2014)
  - If you need prediction intervals or predictive distributions from your linear model, there are also ways of doing that using the bootstrap — see Davison and Hinkley (1997) again, or again Berk (2008), chapter 1
- The view that the optimal linear model is just $\mathrm{Var}\left[\vec{X}\right]^{-1} \mathrm{Cov}\left[Y, \vec{X}\right]$ and all we need is for sample covariance to converge, goes back to Wiener (1949)
  - A notoriously hard-to-read book, but that's because Wiener was trying to explain this in a way which made mathematical sense where instead of regressing Y on a fixed-length vector $\vec{X}$, you're regressing Y on a continuous function, *and* framing the solution in a way which could be implemented using 1940-vintage analog electronics...

Backup: The optimal regression *function*

- We set up the problem

$$\mu = \underset{m:\mathcal{X}\mapsto\mathbb{R}}{\mathrm{argmin}}\, \mathbb{E}\left[\mathbb{E}\left[(Y - m(X))^2 | X\right]\right] \quad (51)$$

$$= \underset{m:\mathcal{X}\mapsto\mathbb{R}}{\mathrm{argmin}} \int dx p(x) \int dy p(y|x)(y - m(x))^2 \quad (52)$$

- For each x, we can minimize the *inner* integral by setting $m(x) = \mathbb{E}[Y|X = x]$
- We just pasted together all of those pointwise minimizers to get μ
- We could do this *because* we were minimizing over all possible functions, so minimizing at x_1 doesn't interfere with minimizing at x_2
- This doesn't work if we limit ourselves to a particular set of functions $\mathcal{M}$
  - Unless of course $\mu \in \mathcal{M}$, i.e., that model is **well-specified**

Backup: Gory details for multivariate predictors

$$m(\vec{X}) = a + \vec{b} \cdot \vec{X} \quad (53)$$

$$(\alpha, \vec{\beta}) = \underset{a,\vec{b}}{\mathrm{argmin}}\, \mathbb{E}\left[(Y - (a + \vec{b} \cdot \vec{X}))^2\right] \quad (54)$$

$$\mathbb{E}\left[(Y - (a + \vec{b} \cdot \vec{X}))^2\right] = \mathbb{E}\left[Y^2\right] + a^2 + \mathbb{E}\left[(\vec{b} \cdot \vec{X})^2\right] \quad (55)$$

$$-2\mathbb{E}\left[Y(\vec{b} \cdot \vec{X})\right] - 2\mathbb{E}[Ya] + 2\mathbb{E}\left[a\vec{b} \cdot \vec{X}\right]$$

$$= \mathbb{E}\left[Y^2\right] + a^2 + \vec{b} \cdot \mathbb{E}\left[\vec{X} \otimes \vec{X}\right] b \quad (56)$$

$$-2a\mathbb{E}[Y] - 2\vec{b} \cdot \mathbb{E}\left[Y\vec{X}\right] + 2a\vec{b} \cdot \mathbb{E}\left[\vec{X}\right]$$

$$(57)$$

Backup: Gory details: the intercept

Take derivative w.r.t. a, set to 0:

$$\begin{aligned} 0 &= -2\mathbb{E}[Y] + 2\beta\mathbb{E}\left[\vec{X}\right] + 2\alpha && (58)\\ \alpha &= \mathbb{E}[Y] - \vec{\beta}\cdot\mathbb{E}\left[\vec{X}\right] && (59)\\ & && (60)\end{aligned}$$

just like when X was univariate

Backup: Gory details: the slopes

$$\begin{aligned} -2\mathbb{E}\left[Y\vec{X}\right] + 2\mathbb{E}\left[\vec{X}\otimes\vec{X}\right]\beta + 2\alpha\mathbb{E}\left[\vec{X}\right] &= 0 && (61)\\ \mathbb{E}\left[Y\vec{X}\right] - \alpha\mathbb{E}\left[\vec{X}\right] &= \mathbb{E}\left[\vec{X}\otimes\vec{X}\right]\beta && (62)\\ \mathbb{E}\left[Y\vec{X}\right] - (\mathbb{E}[Y] - \vec{\beta}\cdot\mathbb{E}\left[\vec{X}\right])\mathbb{E}\left[\vec{X}\right] &= \mathbb{E}\left[\vec{X}\otimes\vec{X}\right]\beta && (63)\\ \mathrm{Cov}\left[Y,\vec{X}\right] &= \mathrm{Var}\left[\vec{X}\right]\beta && (64)\\ \beta &= (\mathrm{Var}\left[\vec{X}\right])^{-1}\mathrm{Cov}\left[Y,\vec{X}\right] && (65)\end{aligned}$$

Reduces to $\mathrm{Cov}[Y,X]/\mathrm{Var}[X]$ when X is univariate

Backup: Gory details: the PCA view

The factor of $\mathrm{Var}\left[\vec{X}\right]^{-1}$ rotates and scales $\vec{X}$ to uncorrelated, unit-variance variables

$$\begin{aligned} \mathrm{Var}\left[\vec{X}\right] &= \mathbf{w}\mathbf{\Lambda}\mathbf{w}^T && (66)\\ \mathrm{Var}\left[\vec{X}\right]^{-1} &= \mathbf{w}\mathbf{\Lambda}^{-1}\mathbf{w}^T && (67)\\ \mathrm{Var}\left[\vec{X}\right]^{-1} &= (\mathbf{w}\mathbf{\Lambda}^{-1/2})(\mathbf{w}\mathbf{\Lambda}^{-1/2})^T && (68)\\ &= \mathrm{Var}\left[\vec{X}\right]^{-1/2}\left(\mathrm{Var}\left[\vec{X}\right]^{-1/2}\right)^T && (69)\\ \vec{U} &\equiv \vec{X}\mathrm{Var}\left[\vec{X}\right]^{-1/2} && (70)\\ \mathrm{Var}\left[\vec{U}\right] &= \mathbf{I} && (71)\\ \vec{X}\cdot\vec{\beta} &= \vec{X}\cdot\mathrm{Var}\left[\vec{X}\right]^{-1}\mathrm{Cov}\left[\vec{X},Y\right] && (72)\\ &= \vec{X}\mathrm{Var}\left[\vec{X}\right]^{-1/2}\left(\mathrm{Var}\left[\vec{X}\right]^{-1/2}\right)^T\mathrm{Cov}\left[\vec{X},Y\right] && (73)\\ &= \vec{U}\mathrm{Cov}\left[\vec{U},Y\right] && (74)\\ & && (75)\end{aligned}$$

Backup: Square root of a matrix

- A square matrix $\mathbf{d}$ is a **square root** of $\mathbf{c}$ when $\mathbf{c} = \mathbf{d}\mathbf{d}^T$
- If there are any square roots, there are many square roots
  - Pick any **orthogonal** matrix $\mathbf{o}^T = \mathbf{o}^{-1}$
  - $(\mathbf{d}\mathbf{o})(\mathbf{d}\mathbf{o})^T = \mathbf{d}\mathbf{d}^T$
  - Just like every real number has two square roots...
- If $\mathbf{c}$ is diagonal, **define** $\mathbf{c}^{1/2}$ as the diagonal matrix of square roots
- If $\mathbf{c} = \mathbf{w}\mathbf{\Lambda}\mathbf{w}^T$, one square root is $\mathbf{w}\mathbf{\Lambda}^{1/2}$

Backup/Aside: R^2 is useless

- $R^2 \equiv \mathrm{Var}\left[\mathrm{fittedvalues}\right]/\mathrm{Var}\left[Y\right]$
- Suppose we *know* $Y = \beta X + \epsilon$, with $\mathrm{Var}\left[\epsilon\right] = \sigma^2$ and ϵ independent of X
- Then $R^2 = \frac{\beta^2 \mathrm{Var}[X]}{\beta^2 \mathrm{Var}[X] + \sigma^2}$
- Consequence 1: R^2 can be arbitrarily close to 0, *even for the completely correct model* (shrink $\mathrm{Var}\left[X\right]$ and/or grow σ^2)
- Consequence 2: If we make $\mathrm{Var}\left[X\right]$ bigger, and the linear model is right, R^2 will grow $\uparrow 1$ (unless $\beta = 0$); R^2 measures the range of the regressor, not goodness of fit
- Suppose the linear model is wrong but we use it anyway; $R^2 = \beta^2 \mathrm{Var}\left[X\right]/\mathrm{Var}\left[Y\right]$
- Consequence 3: R^2 can be arbitrarily close to 1 even if every single assumption of the linear model is badly, obviously broken
- Conclusion: R^2 tells us nothing about how good or bad our model is that we didn't already know from the mean squared error; you'd be better off forgetting about it
- More: Shalizi (2015), chapter 10

References

Berk, Richard A. 2008. *Statistical Learning from a Regression Perspective*. New York: Springer-Verlag.

Buja, Andreas, Richard Berk, Lawrence Brown, Edward George, Emil Pitkin, Mikhail Traskin, Linda Zhao, and Kai Zhang. 2014. "Models as Approximations, Part I: A Conspiracy of Nonlinearity and Random Regressors in Linear Regression." arxiv:1404.1578. http://arxiv.org/abs/1404.1578.

Davison, A. C., and D. V. Hinkley. 1997. *Bootstrap Methods and Their Applications*. Cambridge, England: Cambridge University Press. https://doi.org/10.1017/CBO9780511802843.

Hastie, Trevor, Robert Tibshirani, and Jerome Friedman. 2009. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Second. Berlin: Springer. http://www-stat.stanford.edu/~tibs/ElemStatLearn/.

Shalizi, Cosma Rohilla. 2015. "The Truth About Linear Regression." Online Manuscript. http:///www.stat.cmu.edu/~cshalizi/TALR.

———. n.d. *Advanced Data Analysis from an Elementary Point of View*. Cambridge, England: Cambridge University Press. http://www.stat.cmu.edu/~cshalizi/ADAfaEPoV.

Wiener, Norbert. 1949. *Extrapolation, Interpolation, and Smoothing of Stationary Time Series: With Engineering Applications*. Cambridge, Massachusetts: The Technology Press of the Massachusetts Institute of Technology.