Regularization

In data science
flexible = complex ≅ many parameters

Q: How do we design flexible models which do not overfit the data?

Need to make <u>some</u> assumptions ⟶ assumptions create bias

No free lunch

But, can still control variance in a highly flexible model as follows: Take polynomial regression

$$Y = \sum_{i=0}^{K} \beta_j X^j + \varepsilon$$

Observation: large K values make model flexible, but also result in very erratic behavior when $K \approx N-1$

<u>Regularization</u> is the idea that we will control this erratic behavior by making β_j small

to implement this we return to optimization picture:

$$\underset{\beta}{\text{minimize}} \quad RSS = \underbrace{\sum_{i=1}^{N} (Y_i - \hat{Y}(x_i, D))^2}_{\text{also called } \underline{\text{loss function}}}$$

In regularized regression,

$$\hat{\beta} \text{ minimize } RSS = \sum_{i=1}^{N} (y_i - \hat{y}(x_i, D))^2 + \underbrace{R(\hat{\beta})}$$

Regularization term which generally increases as $\hat{\beta}_j$s increase

In regularized regression we pay a price for larger $\hat{\beta}$

Common Choices

- Ridge regression (Tikhonov regularization) $\quad R(\hat{\beta}) = \lambda \sum_{j=1}^{K} \beta_j^2$

controls how much we weight

- LASSO (least absolute shrinkage and selection operator) $\quad R(\hat{\beta}) = \lambda \sum_{j=1}^{K} |\beta_j|$

Note: Later we will see connection between regularization and incorporating prior knowledge in Bayes's statistics

Regularization reduces variance b.c. less variation in $\hat{\beta}$ from different data sets

... but increases bias b.c making $R(\hat{\beta})$ small often means making $y_i - \hat{y}$ larger

Often want to place different weights on different β_js, this can be achieved by

$$R(\hat{\beta}) = \sum_{j=1}^{K} \lambda_i \beta_j^2$$

in python: OLS_regularized - see section 2 in Colab

A closer look at regularization and the optimization perspective

Let's consider the case of estimating the mean μ from N samples $Y_1, \ldots, Y_N$

(this is like linear regression with $\beta_0 = \mu$, $\beta_1 = 0$!)

In this case: $RSS = \sum_{i=1}^{N} (Y_i - \hat{\mu})^2$

To minimize, solve $\frac{d}{d\mu} RSS = -\sum_{i=1}^{N} 2(Y_i - \hat{\mu}) = 0$

$$\Rightarrow \sum_{i=1}^{N} Y_i = N\hat{\mu} \Rightarrow \hat{\mu} = \bar{Y} \checkmark$$

consistent w/ sample avg./LLN

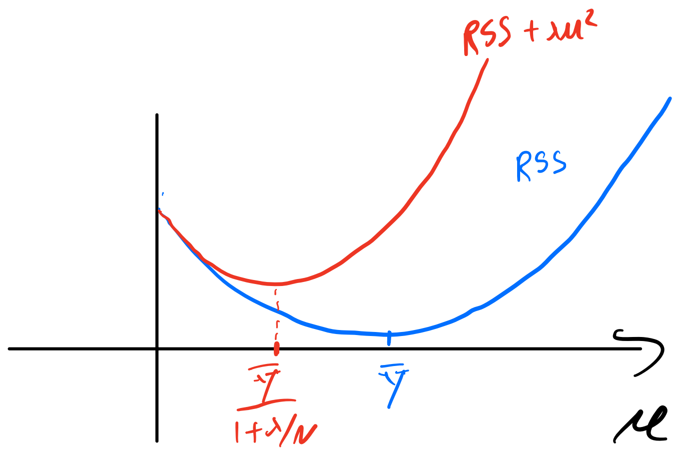
Now minimize $L(\mu) = RSS + \lambda\mu^2$

$$\Rightarrow \frac{d}{d\mu} L(\mu) = -2\sum_{i=1}^{N} (Y_i - \mu) + 2\lambda\mu = 2N\left[\bar{Y} - \mu(1 + \lambda/N)\right] = 0$$

$$\Rightarrow \hat{\mu} = \frac{\bar{Y}}{1 + \lambda/N}$$

larger $\lambda \Rightarrow$ smaller $\hat{\mu}$

more data $\Rightarrow$ regularization matters less



Q: what would happen if we used

$$RSS + \lambda(\mu - \mu_0)^2$$

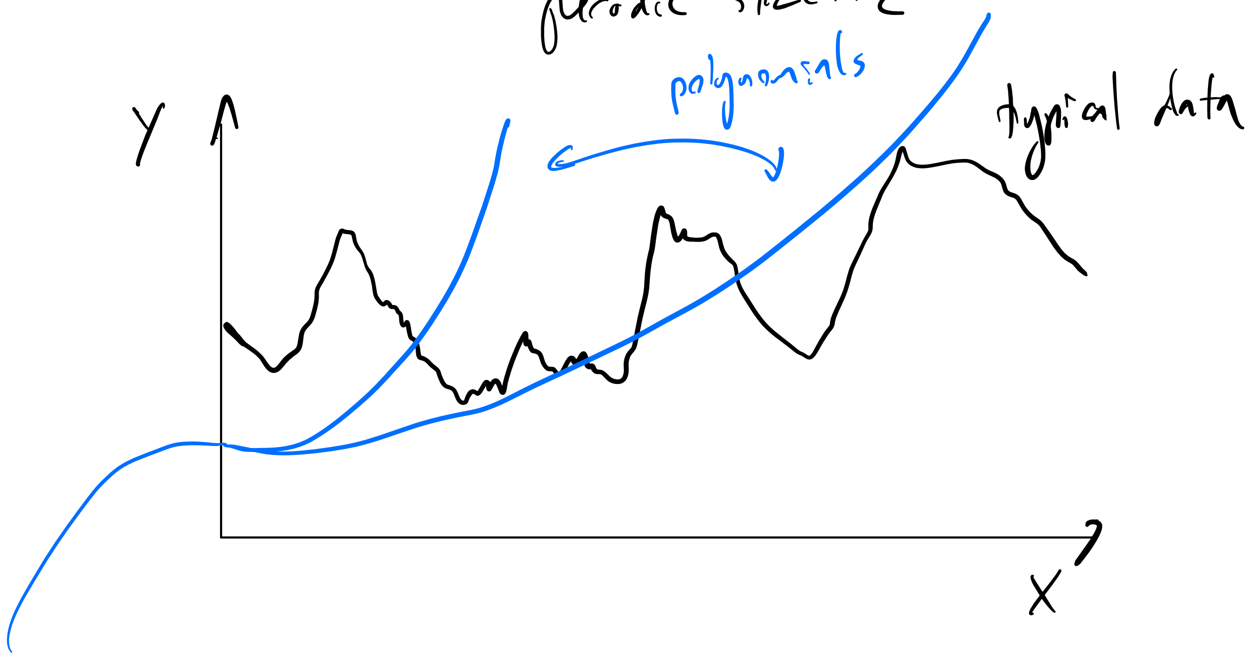
where μ_0 is some fixed $\mathbb{R}$

Fourier and other models

In general, the monomials x^j are not good terms to use in a regression model, particularly when working with time series data.

This is because: (1) X^j and X^i are highly correlated

(2) $X^j \to \infty$ as X grows but often we are dealing w/ data which has some approximately periodic structure



In general, when we build a regression model of the form

$$\underline{Y} = \sum_{j=0}^{K} \beta_j \phi_j(\underline{X}) + \varepsilon$$

we call ϕ_j <u>features</u> or <u>basis functions</u>

Fourier Model

Let's consider data where time measurement are uniformly distributed on some interval $[0, L]$:

$$t \sim \text{Uniform}(0, L)$$

here t will play the same role X has been playing

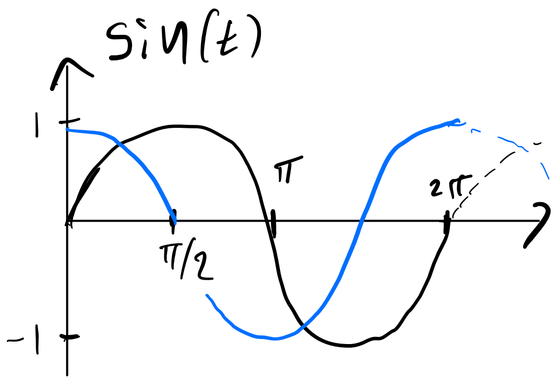
$$Y = \sum_i^l \underbrace{\beta_j \phi_j(t) + \alpha_j \psi_j(t)} + \varepsilon$$

note we could rewrite this in usual linear regression form $\sum_j \beta_j \phi_j$ but it's useful to separate sin and cos terms

where $\phi_j(t) = \sin(2\pi i x / L)$

$\psi_j(t) = \cos(2\pi i x / L)$

Recall:



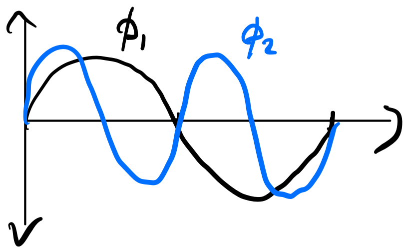
hence ϕ_j, ψ_j are sin/cos functions which go through 1 cycle on $[0, L/i)$

Clearly ϕ_i, ψ_j don't have problem 2

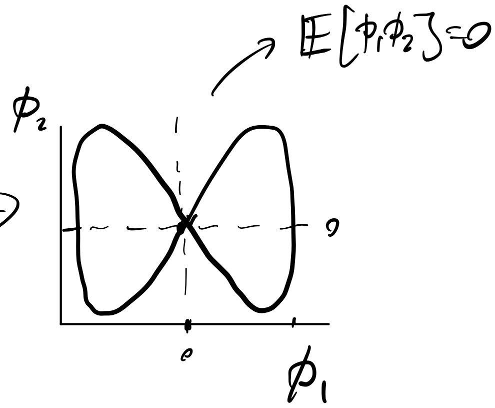
what about problem 1?

Let's look at example:

$$\phi_1 = \sin(2\pi t)$$
$$\phi_2 = \sin(4\pi t)$$



$\Longrightarrow$



Theorem $t \sim \text{Uniform}(0,1)$

then
$$\mathbb{E}[\phi_i(t)\phi_j(t)] = 0 \quad i \neq j$$
$$\mathbb{E}[\phi_i(t)\psi_j(t)] = 0 \quad i \neq j$$
$$\mathbb{E}[\psi_i(t)\psi_j(t)] = 0 \quad i \neq j$$

note that this means Fourier model is very efficient, each basis function contains different information about our data (different freq.)

This means,

$$\beta_i = \frac{\text{cov}(Y, \phi_i)}{\text{var}(\phi_i)}, \quad \psi_i = \frac{\text{cov}(Y, \psi_i)}{\text{var}(\psi_i)}$$

Warning: in practice there will be some correlation between $\phi_i, \phi_j, \psi_i, \psi_j$ due to randomness in sample

However if t points are not random but evenly spaced these correlations vanish!

See examples in colab notebook