ECE 0402 - Pattern Recognition

Lecture 12

ERM: Pick the $h \in \mathcal{H}$ that minimizes $\hat{R}(h)$. This works provided that $\mathcal{H}$ is not "too big", so that we can get a good guarantee of the form

$$R(h) \leq \hat{R}(h) + \epsilon(\mathcal{H}, n)$$

How do we decide how big $\mathcal{H}$ should be?

- One approach to selecting $\mathcal{H}$ would be consider a wide range of $\mathcal{H}$, find the minimal $\hat{R}(h)$ for each one, and to pick the $\mathcal{H}$ that minimizes the VC bound.
  - Unfortunately, the VC bound is very loose so this approach tends to be far too conservative
- Another approach is **structural risk minimization (SRM)** or called **regularization** is to choose a hypothesis by directly minimizing the sum of the training error and some additional term that penalizes the "complexity"

  $$\hat{R}(h) + r(h, n)$$

  SRM differs from ERM approach because:
  - we are considering a much large $\mathcal{H}$, but then penalizing each $h \in \mathcal{H}$ differently
  - **regularizer** $r(h, n)$ is not limited to be some function of VC dimension, but can be much more general (for example it could be tied to some estimate of the variance).
  - Promote "simple hypothesis – if you can get just as small of a training error with a simpler hypothesis, do it!

Linear regression: $f(x) = \beta^T x + \beta_0$ where $\beta \in \mathbb{R}^d,\ \beta_0 \in \mathbb{R}$.

Least squares regression: Select β, β_0 to minimize

$$SSE(\beta, \beta_0) := \sum_{i=1}^{n} (y_i - \beta^T x_i - \beta_0)^2$$

$$\theta = \begin{bmatrix} \beta_0 \\ \beta(1) \\ \vdots \\ \beta(d) \end{bmatrix},\ y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix},\ A = \begin{bmatrix} 1 & x_1(1) & \dots & x_1(d) \\ 1 & x_2(1) & \dots & x_2(d) \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_n(1) & \dots & x_n(d) \end{bmatrix}$$

Then

$$SSE(\theta) = \sum_{i=1}^{n} \left(y_i - \beta^T x_i - \beta_0 \right)^2 = \|y - A\ \theta\|^2$$

The minimizer $\hat{\theta}$ of this quadratic objective function is

$$\hat{\theta} = \left(A^T A \right)^{-1} A^T y$$

this works as long as $A^T A$ is nonsingular. **Overfitting** occurs as the number of features d begins to approach the number of observations n. In this regime we have too many degrees of freedom, we can perfectly interpolate our data. It will be very unstable if we have outliers that doesn't quite fit.

Idea: penalize candidate solutions for using too many features.

One candidate regularizer: $r(\theta) = \|\theta\|^2$ L2 Norm

Regularized version of least-squares regression:

$$\hat{\theta} = \arg\min_{\theta} \|y - A\theta\|^2 + \lambda \|\theta\|^2$$

$\lambda > 0$ is a tuning parameter that controls the trade-off between fit and complexity. This is one example of a more general technique called **Tikhonov regularization**.

$$\hat{\theta} = \arg\min_{\theta} \|y - A\theta\|^2 + \|\Gamma\theta\|^2$$

Note that λ is replaced by the matrix Γ. This makes it more general than fixed λ, Γ allows you to penalize different features differently. **Solution:**

$$\begin{aligned} \|y - A\theta\|^2 + \|\Gamma\theta\|^2 &= (y - A\theta)^T (y - A\theta) + \theta^T \Gamma^T \Gamma \theta \\ &= y^T y + \theta^T A^T A \theta - 2\theta^T A^T y + \theta^T \Gamma^T \Gamma \theta \\ &= y^T y + \theta^T (A^T A + \Gamma^T \Gamma)\theta - 2\theta^T A^T y \end{aligned}$$

$$\begin{aligned} &\frac{\partial}{\partial \theta}\ y^T y + \theta^T (A^T A + \Gamma^T \Gamma)\theta - 2\theta^T A^T y \\ &\quad = 2(A^T A + \Gamma^T \Gamma)\theta - 2A^T y \end{aligned}$$

setting this equal to 0 and solving for θ yields,

$$\hat{\theta} = (A^T A + \Gamma^T \Gamma)^{-1} A^T y$$

Suppose $\Gamma = \sqrt{\lambda}\mathcal{I}$, then

$$\hat{\theta} = (A^T A + \lambda \mathcal{I})^{-1} A^T y$$

We can use Lagrange multipliers (KKT conditions) to show that

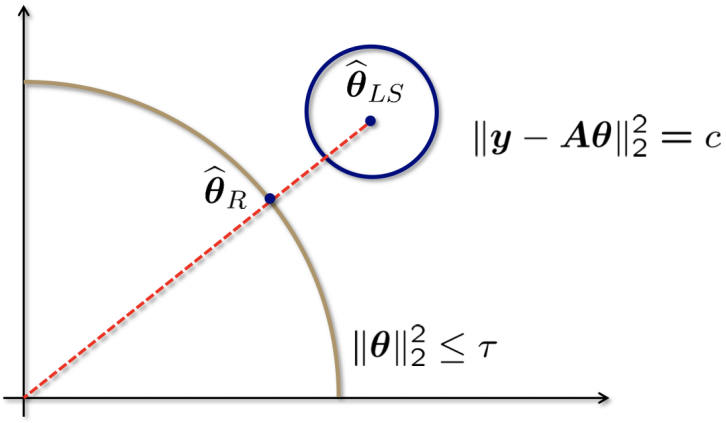
$$\hat{\theta} = \arg\min_{\theta} \|y - A\theta\|^2 + \|\Gamma\theta\|^2$$

is formally equivalent to

$$\hat{\theta} = \arg\min_{\theta} \|y - A\theta\|^2$$
$$\text{subject to} \quad \|\Gamma\theta\|^2 \leq \tau$$

for a suitable choice of τ.

Tikhonov versus least squares Assume $\Gamma = \mathcal{I}$ which makes $\|\theta\|^2 \leq \tau$



All Tikhonov regularization is doing, taking this least squares estimate and shrinking it towards the origin. If you didn't have Γ identity, it is not shrinking every coordinate by an equal amount, but the general idea is that it is going to shrink the solution toward the origin. That's why, sometimes you may of seen Tikhonov regularization under the general type of "shrinkage estimators". Shrinkage estimators are estimators that "shrink" the naïve estimate towards some implicit guess.

Example: $x_1, ..., x_n$ are i.i.d. samples of an unknown distribution, how can we estimate the variance?

$$\hat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2 \implies \mathbb{E}[\hat{\sigma}^2] = \frac{n-1}{n}\sigma^2$$

This is a biased estimator (it shrinks slightly towards zero). However, it also achieves a lower MSE than the unbiased estimate. This actually happens more than you think, so variance example is not a cooked up one. A shrinkage type estimator will actually perform much better than a naïve approach.

Have you heard **Stein's paradox** (1955) ? (One reference for a quick read: `http://www.statslab.cam.ac.uk//~rjs57/SteinParadox.pdf`). He proved this idea of shrinkage is super important.

Consider the simplest estimation problem where you observe $y = \theta + n$, n is i.i.d. Gaussian noise. How to estimate θ?

- Obvious estimate, $\hat{\theta} = y$. How could you possible do any better than that?
- If the dimension is 3 or higer, then this is suboptimal in terms of MSE: $\mathbb{E}[\left\|\hat{\theta} - \theta\right\|]$
- You can do better by shrinking towards **any** guess for θ! If you provide any guess for θ and set-up a regularized problem that will end up doing no worse. And if the guess is pretty good, it will do a lot better.
  - people usually shrink towards the origin
  - a better guess leads to bigger improvements

Ridge Regression:

In the context of regression, Tikhonov regularization has a special name: ridge regression.

Ridge regression is what we are talking about here, but with a very special choice of Γ:

$$\Gamma = \begin{bmatrix} 0 & 0 & 0 & ... & 0 \\ 0 & \sqrt{\lambda} & 0 & ... & 0 \\ 0 & 0 & \sqrt{\lambda} & ... & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & ... & \sqrt{\lambda} \end{bmatrix}$$

This choice makes sense in the context of regression. Remember, in regression, β_0 offset parameter was int he first entry (and then vector β of all of the slopes in the remainder). We are penalizing all coefficients in β equally, but not penalizing the offset β_0.

There are bunch of other choices of regularizers. Here, we primarily talked about the idea of using the Euclidean norm of our vector of coefficients as a good regularizer, but there are lots of other choices.

- Akaike information criterion (AIC)
- Bayesian information criterion (BIC)

$$r(\theta) \approx \|\theta\|_0 := |\text{supp}(\theta)|$$

  This is a very natural one but unfortunately basically almost impossible to work with on big problems. Unfortunately if you have a lot of features, optimizing over this criterion leads to non-convex optimization problems that are NP-hard to optimize...

- Least absolute shrinkage and selection operator (LASSO)

$$r(\theta) = \|\theta\|_1 = \sum_j |\theta(j)|$$

  - also results in shrinkage, but where all coordinates are shrunk by the same amount
  - promotes sparsity
  - can think of $\|\theta\|_1$ as a more computationally tractable replacement for $\|\theta\|_0$.

We can applies these ideas to other regression/classification problems. For example in logistic regression where we were doing MLE by minimizing the log-likehood, here we can replace

$$min_\theta \ -l(\theta)$$

with

$$min_\theta \ -l(\theta) + \lambda\|\theta\|^2$$

This has a similar interpretation to the least squares regularization.

- makes the Hessian matrix well conditioned (not very much data compared to the dimension when you use Newton's method)
- in general, it super useful when the number of observations are small

The regularization idea can also give us a new way to think about designing linear classifiers. For example when we find (w, b) by minimizing training error,

$$\frac{1}{n}\sum_{i=1}^{n} 1_{y_i(w^T x_i + b)<0}$$

This is actually much harder than it looks and is not computationally tractable for large problems. Instead we can consider replacing the indicator with

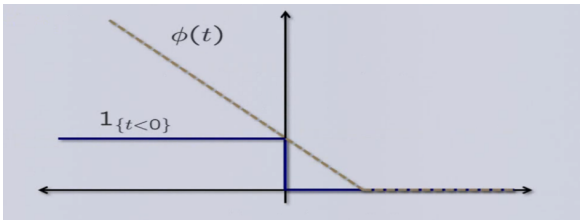
$$\frac{1}{n}\sum_{i=1}^{n} \phi(y_i(w^T x_i + b))$$

where $\phi(t)$ is some upper bound on $1_{t<0}$. One standard upper bound in literature is **"hinge loss"**:

$$\phi(t) = max\{0, 1-t\} := (1-t)_+$$

Let's try to minimize

$$\frac{1}{n}\sum_{i=1}^{n}(1 - y_i(w^T x_i + b))_+$$



but to prevent overfitting, let's add a regularization penalty to w:

$$min_{w,b}\ \frac{1}{n}\sum_{i=1}^{n}(1-y_i(w^Tx_i+b))_+ + \frac{\lambda}{2}\|w\|^2$$

Does this a bit look like optimal **soft-margin hyperplane**?

$$\begin{aligned} min_{w,b,\xi}\ \ & \frac{1}{2}\|w\|^2 + \frac{C}{n}\sum_{i=1}^{n}\xi_i \\ \text{s.t.}\quad & y_i(w^Tx_i+b) \geq 1-\xi_i \quad i=1,...,n \\ & \xi_i \geq 0 \quad i=1,...,n \end{aligned}$$

If $C=\frac{1}{\lambda}$, these two optimization problems are solved by same w, b. Since scaling an objective function by a positive constant does not change the solution, it suffices to show that the same w, b solve both.

$$\text{P1}:\ min_{w,b}\ \frac{1}{2}\|w\|^2 + \frac{1}{n\lambda}\sum_{i=1}^{n}(1-y_i(w^Tx_i+b))_+$$

and

$$\begin{aligned} \text{P2}: min_{w,b,\xi}\ \ & \frac{1}{2}\|w\|^2 + \frac{1}{\lambda n}\sum_{i=1}^{n}\xi_i \\ \text{s.t.}\quad & y_i(w^Tx_i+b) \geq 1-\xi_i \quad i=1,...,n \\ & \xi_i \geq 0 \quad i=1,...,n \end{aligned}$$

Suppose (w^*, b^*) optimize P1, then (w^*, b^*, ξ^*) optimize P2 where $\xi^* = max(0, 1-y_i(w^{*T}x_i+b^*))$. Now time for my favorite step, proof by contradiction: suppose that this is not true and let (w, b, ξ) be the true solution to QP2.

- If $\xi_i > 0$, then we must have that $y_i(w^Tx_i+b) = 1-\xi_i$, since otherwise we could decrease the objective without violating any constraints
- If $\xi_i = 0$, then $y_i(w^Tx_i+b) \geq 1$
- Thus,

$$\sum_{i=1}^{n}\xi_i = \sum_{i=1}^{n}(1-y_i(w^Tx_i+b))_+$$

  This is the sum of hinge loss functions.

Hence, we can re-write P1 as,

$$
\begin{aligned}
min_{w,b}\ & \frac{1}{2}\|w\|^2 + \frac{1}{n\lambda}\sum_{i=1}^{n}(1 - y_i(w^T x_i + b))_+ \\
&= \frac{1}{2}\|w\|^2 + \frac{1}{n\lambda}\sum_{i=1}^{n}\xi_i \\
&< \frac{1}{2}\|w^*\|^2 + \frac{1}{n\lambda}\sum_{i=1}^{n}\xi_i^* \\
&= \frac{1}{2}\|w^*\|^2 + \frac{1}{n\lambda}\sum_{i=1}^{n}(1 - y_i(w^{*T} x_i + b^*))_+
\end{aligned}
$$

this contradicts the optimality of (w^*, b^*) for P1.

Next: We must somehow determine an appropriate value for λ only using the training data. The problem of selecting the "free parameters" is often called **model selection** in learning set-up and is a very critical step in most practical learning scenarios.