Lecture 17 4/7/21

Response Space	Types of Modeling	g return	Example Alg.
$\mathcal{Y} \subseteq \mathbb{R}$	regression	$\hat{y} \in \mathcal{Y}$	OLS
$\mathcal{Y} = \{c_1, c_2, \ldots, c_K\}$	classification	$\hat{y} \in \mathcal{Y}$	KNN
if $K=2$, $\mathcal{Y} = \{\overset{c_1}{0}, \overset{c_2}{1}\}$	binary classification	$\hat{y} \in \mathcal{Y}$	SVM
$\mathcal{Y} \subseteq \mathbb{R}_{\geq 0}$	Survival	$\hat{y} \in \mathcal{Y}$	Weibull regression
$\mathcal{Y} \in \{0, 1, 2, \ldots\}$	Count	$\hat{y} \in \mathcal{Y}$	Poisson Regression
$\mathcal{Y} \in (0, 1)$	proportion	$\hat{y} \in \mathcal{Y}$	Beta regression
$\mathcal{Y} = \{c_1, c_2, \ldots, c_K\}$	probability estimation	$\hat{\vec{p}} := P(Y = c_1 \mid \vec{x})$, $P(Y = c_2 \mid \vec{x})$, $\vdots$, $P(Y = c_K \mid \vec{x})$	Multilayer regression
$\mathcal{Y} \in \{c_1, c_2, \ldots, c_K\}$ ordinal	probability estimation		Proportional odds model
$K=2$, $\mathcal{Y} = \{\overset{c_1}{0}, \overset{c_2}{1}\}$	probability estimation	$\hat{p} : P(Y = 1 \mid \vec{x})$	Logistic regression

---

If $y = \{0, 1\}$ for all i,

$$
\begin{aligned}
y &= t(\vec{z}) \\
&= f(\vec{z}) + \delta \text{ where } \delta \in \{0, -1, +1\} \\
&= h^*(\vec{x}) + \mathcal{E} \text{ where } \mathcal{E} \in \{0, -1, +1\} \\
&= g(\vec{x}) + e \text{ where } e \in \{0, -1, +1\}
\end{aligned}
$$

$\Longleftrightarrow Y \sim \text{Bern}(t(\vec{z}))$

$$
e \quad \begin{array}{cc|c|c}
 & & \multicolumn{2}{c}{g(\vec{x})} \\
 & & 0 & 1 \\
\hline
Y & 0 & 0 & -1 \\
\hline
 & 1 & +1 & 0
\end{array}
$$

False positive (-1)

False negative ($+1$)

How do we build a probability estimation model? Naively, $g_0 = \bar{Y}$

We now view Y as a realization from a random variable (bernoulli). We will assume there exists a function $f_{pr}(\vec{x}): \mathbb{R}^{p+1} \to (0,1)$ and this function is the best guess of the probability $P(Y=1 \mid \text{xvec})$ you can create with xvec.

$$Y \sim \text{Bern}\Big(f_{pr}(\vec{x}) + \underbrace{t(\vec{z}) - f_{pr}(\vec{x})}_{\delta_{pr}}\Big)$$

$\Rightarrow \underbrace{Y \sim \text{Bern}(f_{pr}(\vec{x}))}$. f_{pr} is the model we want to find.

Let's assume that all the data (all the n observations) in D are $\underbrace{\text{independently}}$ realized

$$P(D) = P(Y_1 = y_1, Y_2 = y_2, \ldots, Y_n = y_n \mid \vec{x}_1, \ldots, \vec{x}_n)$$

$$= \prod_{i=1}^{n} P(Y_i = y_i \mid \vec{x}_i)$$

$$= \prod_{i=1}^{n} f_{pr}(\vec{x}_i)^{y_i}(1 - f_{pr}(\vec{x}_i))^{1-y_i}$$

$V \sim \text{Bern}(\theta)$
$\quad \shortparallel$
241 Review $\theta^v(1-\theta)^{1-v}$

Now we want to "fit" f_{pr} using our data (learning from data paradigm). How? Is it even possible? NO. We cannot fit arbitrary functions in any dimension. We need a set of candidate functions that we can fit. Call that $\mathcal{H}_{pr}$. Each element in this set maps $\mathbb{R}^{p+1} \to (0,1)$. How about:

$$\mathcal{H}_{pr} = \{\vec{w} \cdot \vec{x} : \vec{w} \in \mathbb{R}^{p+1}\}\ ?$$

We can't use this since it returns values outside $(0,1)$, the space of legal probabilities. But... we really like wvec · xvec because (1) easy to interpret and we have lots of intuition about it from all of our previous modeling we've done and (2) monotonic in each of the x_j's. How do we have our cake and eat it too? We need a function that takes wvec · xvec and maps it into the space $(0,1)$, i.e. $\phi: \mathbb{R} \to (0,1)$ which is called a "link function" I think because it links the two spaces (the reals and the probs). We restrict the link function to be strictly increasing. Thus,

$$\mathcal{H}_{pr} = \{\phi(\vec{w} \cdot \vec{x}) : \vec{w} \in \mathbb{R}^{p+1}\}$$

These types of models are called "generalized linear models" (glm) because they retain wvec · xvec (the linear model) but then manipulate it in some way. Which link function should we use? There are three common ones. In order of use:

① Logistic / logit: $\phi(u) := \dfrac{e^u}{1+e^u} = \dfrac{1}{1+e^{-u}}$. Note: $1 - \phi(u) = \dfrac{1}{1+e^u}$

② Probit: $\phi(u) := F_Z(u)$ i.e. the CDF of the std. normal.

③ Complementary Log-Log (cloglog)

$$\phi(u) = 1 - e^{-e^u} \Rightarrow 1 - \phi(u) = e^{-e^u} \Rightarrow \ln(1-\phi(u)) = -e^u$$

$$\Rightarrow -\ln(1-\phi(u)) = e^u \Rightarrow u = \overbrace{\ln(-\ln(\underbrace{1-\phi(u)}_{\text{complement}})}^{\text{log-log}}$$

Let's employ the logistic link function:

$$\mathcal{H} = \left\{ \frac{1}{1+e^{-\vec{w}\cdot\vec{x}}} : \vec{w} \in \mathbb{R}^{p+1} \right\}$$

What is $\mathcal{A}$? How to get $g \in \mathcal{H}$?

Why not find the wvec that provides us the highest probability?

$$\mathcal{A} : \vec{b} := \underset{\vec{w} \in \mathbb{R}^{p+1}}{\text{argmax}} \left\{ \underbrace{\prod_{i=1}^{n} \left(\frac{1}{1+e^{-\vec{w}\cdot\vec{x}}}\right)^{y_i} \left(\frac{1}{1+e^{\vec{w}\cdot\vec{x}}}\right)^{1-y_i}}_{P(D)} \right\}$$

In OLS, we took the derivative and set it equal to zero to solve for bvec and we found an analytical solution. However, there is no analytical solution here. You need to use a computer.

$$\vec{\nabla} P(D) \overset{set}{=} \vec{0}_{p+1} \text{ and approximate}$$

Usually this is done with "gradient descent". Computing bvec is called "running a logistic regression". Once this is done ... we can predict using

$$\hat{p} = g_{pr}(\vec{x}) = \phi(\vec{b}\cdot\vec{x}) = \frac{1}{1+e^{-\vec{b}\cdot\vec{x}}} \text{ hopefully close to } f_{pr}(\vec{x})$$

$$\overset{\cdot\cdot}{=}$$

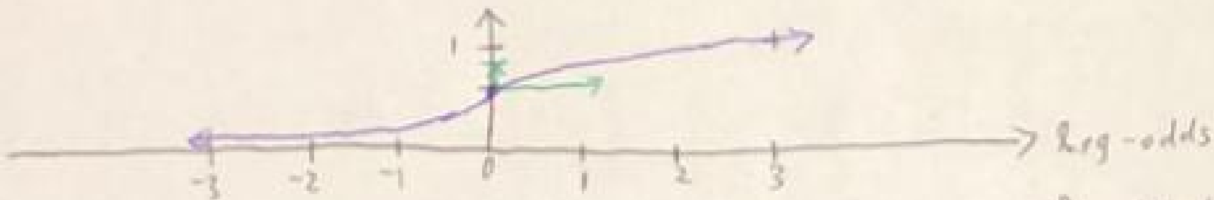
$$\hat{p}(Y=1 \mid \vec{x})$$

What is the interpretation of the slope coefficients (the entries in the b-vec)?

$$\hat{p} = \frac{1}{1+e^{-\vec{b}\cdot\vec{x}}} \Rightarrow \frac{1}{\hat{p}} = 1+e^{-\vec{b}\cdot\vec{x}} \Rightarrow \frac{1}{\hat{p}} - 1 = e^{-\vec{b}\cdot\vec{x}} \Rightarrow \frac{1-\hat{p}}{\hat{p}} = e^{-\vec{b}\cdot\vec{x}} \Rightarrow \ln\left(\frac{1-\hat{p}}{\hat{p}}\right) = -\vec{b}\cdot\vec{x}$$

$$\Rightarrow \underbrace{\ln\Big(\underbrace{\frac{\hat{p}}{1-\hat{p}}}_{\text{odds}}\Big)}_{\text{log-odds}} \qquad \text{odds} = \frac{\hat{p}}{1-\hat{p}}$$

$\Rightarrow b_j$ is the ~~changing~~ change in the log-odds of $Y=1$ if x_j increases by 1.



$0 = \ln\left(\frac{\hat{p}}{1-\hat{p}}\right) \Rightarrow \phi(0) = \frac{1}{1+e^{-0}} = \frac{1}{1+1} = 0.5$

log-odds	prob
0	0.5
-1	0.27 ≈ 1/4
+1	0.73 ≈ 3/4
-2	0.12 ≈ 1/8
+2	0.88 ≈ 7/8
⋮	
$-\infty$	0
$+\infty$	1

The logistic link function is highly nonlinear.

0 → 1 logodds ⟺ 50% → 73% in prob.

3 → 4 logodds ⟺ 95% → 98% in prob.

Probability estimation models predict probabilities but we observe labels (i.e. 0 or 1). The true probabilities f_{pr} are unobserved! We need a metric called a "scoring rule" S that can compare a $\hat{p}$ value to a y value.

A "proper scoring rule" $S(\hat{p}, y)$ is one where:

$$\forall_i \quad f_{pr}(\vec{x}_i) = \text{argmax}\{S(\hat{p}_i, y_i)\}$$

We will study two proper scoring rules:

① Brier score (1950). Let $s_i := -(y_i - \hat{p}_i)^2 \le 0$

$$\bar{s} := \frac{1}{n}\sum_{i=1}^{n} s_i \le 0$$

② Log scoring rule. Let $s_i := y_i \ln(\hat{p}_i) + (1 + y_i)\ln(1-\hat{p}_i) \le 0$

$$\bar{s} = \frac{1}{n}\sum s_i \le 0$$

These scores are used as an "R^2" of the model (but they're not between 0 and 1) in a conceptual sense. The closer to zero, the better the probability estimation model.