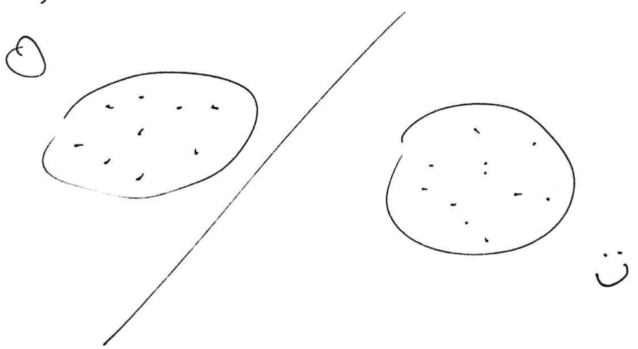
Generative Models

Sofar, Looked At

$p(y \mid x ; \theta)$

For classification, really

i) A straight line to distinguish classes



Can do this differently

Model ○ $\Rightarrow$ Can check X performance under Both Models

Model ☺

Basically Learning $p(X \mid y)$ and $p(y)$ instead of $p(y \mid x)$

These are called generative (Not discriminative) models

If y is class membership of ○ or ☺ then

$p(x \mid y = 0)$

Have $P(y)$, $P(X|Y)$, then we can use $\boxed{\text{Bayes' Theorem}}$ to calculate $P(y|x)$.

$$P(y|x) = \frac{P(x|y)P(y)}{P(x)}$$

$$P(x) = P(x|y=1)P(y=1) + P(x|y=0)P(y=0).$$

Thus, $$\arg\max_y P(y|x) = \arg\max_y \frac{P(x|y)P(y)}{P(x)} = \arg\max_y P(x|y)P(y)$$

Start with $\boxed{\text{Gaussian Discriminant Analysis (GDA)}}$; Need $\boxed{\text{Multivariate Normal Distributions.}}$

In d dimensions, parameterized by mean vector $\mu \in \mathbb{R}^d$ and <u>covariance matrix</u> $\Sigma \in \mathbb{R}^{d \times d}$

with $\Sigma \geq 0$ symmetric and positive semidefinite

$$\left(z^T \Sigma z \geq 0, \quad \forall z \neq 0\right), \text{ then}$$
$\qquad\qquad\qquad\qquad\qquad\qquad\ \ \in \mathbb{R}^d$

$$P(x; \mu, \Sigma) = \frac{1}{(2\pi)^{d/2} \underbrace{|\Sigma|}_{\text{Determinant}}^{1/2}} \exp\left(-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)\right)$$

Note for $X \sim N(\mu, \Sigma)$

(vs μ, σ for 1D Normal),

$E[X] = \mu$.

The covariance for a vector-valued RV X is

$$\mathrm{Cov}(X) = E\left[(X - E[X])(X - E[X]^T)\right]$$
$$= E[XX^T] - (E[X])(E[X])^T,$$

for $X \sim N(\mu, \Sigma)$, $\mathrm{cov}(X) = \Sigma$

Gaussian Discriminant Analysis

Want to classify in situation where inputs are features X, continuous random variable

Model $P(X|y)$ as multivariate normal

i.e. $y \sim \mathrm{Bernoulli}(\varphi)$

$$\left.\begin{aligned} x|y=0 &\sim N(\mu_0, \Sigma) \\ x|y=1 &\sim N(\mu_1, \Sigma) \end{aligned}\right\} \begin{array}{l}\text{Same Covariance} \\ \text{Difference Mean}\end{array}$$

$$p(y) = \phi^{y}(1-\phi)^{1-y}$$

$$p(x|y=0) = \frac{1}{(2\pi)^{1/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu_0)\Sigma^{-1}(x-\mu_0)\right)$$

$$p(x|y=1) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu_0)\Sigma^{-1}(x-\mu_0)\right)$$

And log likelihood:

$$\ell(\varphi, \mu_0, \mu_1, \Sigma) = \log \prod_{i=1}^{n} p(x^{(i)}, y^{(i)}; \varphi, \mu_0, \mu_1, \Sigma)$$

$$= \log \prod_{i=1}^{n} p(x^{(i)} | y^{(i)}; \mu_0, \mu_1, \Sigma)\, p(y^{(i)}; \varphi)$$

$$\varphi = \frac{1}{n} \sum_{i=1}^{n} 1\{y^{(i)} = 1\}$$

$$\mu_0 = \frac{\sum_{i=1}^{n} 1\{y^{(i)} = 0\}\, x^{(i)}}{\sum_{i=1}^{n} 1\{y^{(i)} = 0\}}$$

$$\mu_1 = \frac{\sum_{i=1}^{n} 1\{y^{(i)} = 1\} x^{(i)}}{\sum_{i=1}^{n} 1\{y^{(i)} = 1\}}$$

$$\Sigma = \frac{1}{n} \sum_{i=1}^{n} \left(x^{(i)} - \mu_{y^{(i)}}\right)\left(x^{(i)} - \mu_{y^{(i)}}\right)^T$$

$$\begin{aligned} l(\varphi, \mu_0, \mu_1, \Sigma) = \sum_{i=1}^{n} \Big(& y^{(i)} \log \phi + (1-y^{(i)}) \log(1-\phi) \\ & - \frac{1}{2} \log|\Sigma| + C \\ & + (x^{(i)} - \mu)^T \Sigma^{-1} (x^{(i)} - \mu) \Big) \end{aligned}$$

Depends on $y^{(i)}$ if $= 0$

$\Rightarrow \mu = \mu_0 = 1 \Rightarrow \mu = \mu_1$

Take partials wrt $\phi, \mu_0, \mu_1, \Sigma$; set $= 0$:

$$\frac{\partial l}{\partial \varphi} = \sum_{i=1}^{n} \frac{y^{(i)}}{\varphi} - \sum_{i=1}^{n} \frac{(1-y^{(i)})}{1-\varphi} = 0$$

$$\Rightarrow (1-\phi) \sum_{i=1}^{n} y^{(i)} = \varphi \sum_{i=1}^{n} 1 - y^{(i)}$$

$$\sum_{i=1}^{n} y^{(i)} - \phi \sum_{i=1}^{n} y^{(i)} = n\varphi - \varphi \sum_{i=1}^{n} y^{(i)}$$

$$\varphi = \frac{1}{n} \sum_{i=1}^{n} y^{(i)} = \frac{1}{n} \sum_{i=1}^{n} 1\{y^{(i)} = 1\}$$

Note $\nabla_A \operatorname{tr} ABA^TC = CAB + C^TAB^T$, Σ^{-1} is symmetric

$$\nabla_{\mu_0} l = \nabla_{\mu_0} \sum_{i=1}^{n} \operatorname{tr} \underbrace{(x^{(i)} - \mu)^T \Sigma^{-1} (x^{(i)} - \mu)}_{1 \times 1} I$$

$$= - \sum_{i=1}^{n} \left[I (x^{(i)} - \mu_0)^T \Sigma^{-1} + I (x^{(i)} - \mu_0)^T (\Sigma^{-1})^T \right] 1\{y^{(i)} = 0\}$$

$$= 0$$

$$\Rightarrow -2\Sigma^{-1} \underbrace{\boxed{\sum_{i=1}^{n} (x^{(i)} - \mu_0) 1\{y^{(i)} = 0\}}}_{=0} = 0 \qquad \Sigma^{-1} \text{ is invertible}, \neq 0$$

$$\Rightarrow \sum_{i=1}^{n} (x^{(i)} - \mu_0) 1\{y^{(i)} = 0\} = 0$$

$$\Rightarrow \mu_0 = \frac{\sum_{i=1}^{n} 1\{y^{(i)} = 0\} x^{(i)}}{\sum_{i=1}^{n} 1\{y^{(i)} = 0\}}$$

Now, find Σ: Note $\frac{1}{|\Sigma|} = |\Sigma^{-1}|$

Instead of $x^{(i)}$, write x, μ instead of μ_0 or μ_1.

$\mathrm{tr}\, ABC = \mathrm{tr}\, CAB$

Take the derivative (Dropping Terms That Don't Matter)

$$l = \frac{n}{2} \log |\Sigma^{-1}| - \frac{1}{2} \sum_{i=1}^{n} \mathrm{tr} (x-\mu)(x-\mu)^T \Sigma^{-1}$$

$$l(\varphi, \mu_0, \mu_1, \Sigma) = \sum_{i=1}^{n} \Big[y^{(i)} \log \varphi + (1-y^{(i)}) \log(1-\varphi)$$

$$- \frac{1}{2} \log|\Sigma| + C + (x^{(i)} - \mu_{y^{(i)}})^T \Sigma^{-1} (x^{(i)} - \mu_{y^{(i)}}) \Big]$$

Ignoring terms that go to 0 when we take gradient.

$$l = \frac{n}{2} \log|\Sigma^{-1}| - \frac{1}{2} \sum_{i=1}^{n} tr\,(x - \mu_{y^{(i)}})(x - \mu_{y^{(i)}})^T \Sigma^{-1}$$

Note $\nabla_A \log|A| = (A^{-1})^T$, $\nabla_B \, tr AB = A^T$.

Let $\Lambda = \Sigma^{-1}$, then

$$\nabla_\Lambda l = \nabla_\Lambda \Big[\frac{n}{2} \log|\Lambda| - \frac{1}{2} \sum_{i=1}^{n} tr\,(x-\mu)(x-\mu)^T \Lambda \Big]$$

$$= \frac{n}{2} \Lambda^{-1} - \frac{1}{2} \sum_{i=1}^{n} (x-\mu)(x-\mu)^T \overset{\text{set}}{=} 0$$

for critical point

$$n \Lambda^{-1} = \sum_{i=1}^{n} (x-\mu)(x-\mu)^T$$

$$n \Sigma = \sum_{i=1}^{n} (x-\mu)(x-\mu)^T$$

$$\Rightarrow \Sigma = \frac{1}{n} \sum_{i=1}^{n} (x - \mu_{y^{(i)}})(x - \mu_{y^{(i)}})^T$$

Look at the following as function of x:

$$P(y=1 \mid x ; \phi, \mu_0, \mu_1, \Sigma) = \frac{1}{1+\exp(-\theta^T x)}$$

This looks like logistic regression!

Thus, why would I use $\boxed{\text{GDA}}$ vs. $\boxed{\text{Log. Reg.}}$ in any particular setting?

Give different outputs

Turns out, $P(x|y)$ multivariate Gaussian w/ Σ $\Rightarrow$

$P(y|x)$ is logistic.

But $P(y|x)$ logistic $\not\Rightarrow$ $P(x|y)$ mult. Gaussian,

$$\left.\begin{array}{l} x|y=0 \sim \text{Poisson}(\lambda_0) \\ x|y=1 \sim \text{Poisson}(\lambda_1) \end{array}\right\} \Rightarrow P(y|x) \text{ Logistic}$$

If your data looks $\approx$ Gaussian,

then GDA will work better.

Logistic Reg. cares less about underlying distributions, so is more robust.

Naive Bayes

Recall: <u>Sum Rule</u>: $P(x) = \sum_y P(x,y)$

<u>Product Rule</u>: $P(x,y) = P(y|x)P(x)$

Want a model that assumes X is discrete and y is discrete classification.

Going to introduce the "Naive Bayes" model for binary classification.

Sample problem - SPAM filtering

Think of an email x as a vector of whether or not a word appears in an email:

$$x = \begin{array}{c} \\ \left[\begin{array}{c} 0 \\ 1 \\ 1 \\ 0 \\ 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{array}\right] \end{array} \left.\begin{array}{l} \text{win} \\ \text{pure} \\ \text{gold} \\ \text{xxx} \\ \text{office} \\ \text{hours} \\ \text{copier} \\ \vdots \\ \text{toast} \end{array}\right\} \text{words called vocabulary} \in \mathbb{R}^d$$

d is # of words

Goal is to model $p(x|y)$, but if d is large size of x space goes like is 2^d.

Model this w/ multinomial need $2^d - 1$ parameters.

Thus, assume, to reduce size of model, that x_i's are conditionally independent given y (Not Ind.). This means if I already know what class an email is in, then appearance of individual words is independent.

i.e. if I know it is spam, knowing "pure" occurs does NOT tell me anything about the occurence of "gold".

Mathematically, $P(x_7|y) = P(x_7|y, x_3)$

$$\overset{\text{Not}}{\Rightarrow} P(x_7) = P(x_7|x_3)$$

How is this helpful?

$$P(x_1, \ldots, x_d \mid y) = P(x_1 \mid y)\, P(x_2 \mid y, x_1) \cdot \ldots \cdot P(x_d \mid y, x_1, \ldots, x_{d-1})$$

Always True

Naive Bayes' Assumption

$$= P(x_1 \mid y)\, P(x_2 \mid y) \cdot \ldots \cdot P(x_d \mid y)$$

$$= \prod_{j=1}^{d} P(x_j \mid y)$$

What are the learnable parameters?

$$\varphi_{j|y=1} = P(x_j = 1 \mid y = 1), \quad \varphi_y = P(y=1)$$

$$\varphi_{j|y=0} = P(x_j = 1 \mid y = 0)$$

For training set $\{(x^{(i)}, y^{(i)}),\ i = 1, \ldots, n\}$

$$\mathcal{L}(\varphi_y, \varphi_{j|y=0}, \varphi_{j|y=1}) = \prod_{i=1}^{n} P(x^{(i)}, y^{(i)})$$

Maximized w/ ..

Maximized with

$$\varphi_{j|y=1} = \frac{\sum_{i=1}^{n} 1\{x_j^{(i)} = 1 \wedge y^{(i)} = 1\}}{\sum_{i=1}^{n} 1\{y^{(i)} = 1\}}$$

$$\varphi_{j|y=0} = \frac{\sum_{i=1}^{n} 1\{x_j^{(i)} = 1 \wedge y^{(i)} = 0\}}{\sum_{i=1}^{n} 1\{y^{(i)} = 0\}}$$

$$\varphi_y = \frac{\sum_{i=1}^{n} 1\{y^{(i)} = 1\}}{n}$$

Given a new email, how do I classify it?

Calculate $P(y=1 \mid x)$, $P(y=0 \mid x)$.

Choose Bigger One

$$P(y=1 \mid x) = \frac{P(x \mid y=1)\,P(y=1)}{P(x)}$$

$$= \frac{\left(\prod_{i=1}^{d} P(x_i \mid y=1)\; P(y=1)\right)}{\left(\prod_{j=1}^{d} P(x_j \mid y=1)\right) P(y=1) + \left(\prod_{j=1}^{d} P(x_j \mid y=0)\right) P(y=0)}$$

What if Naive Bayes nevers saw a word in the training data "pure"

$$\varphi_{j|y=0} = \varphi_{j|y=1} = 0$$

Now this word appears in an email you want to classify $\Rightarrow$ $p(y=1 \mid x) = \frac{0}{0}$. What to do?

For estimating a multinomial distribution $\varphi_j = P(z=j)$ with n observations and k classes $\{z^{(1)}, \ldots, z^{(n)}\}$, max likelihood is

$$\varphi_j = \frac{\sum_{i=1}^{n} 1\{z^{(i)} = j\}}{n}$$ to not allow $\varphi_j = 0$.

Use $\boxed{\text{Laplace Smoothing}}$:

$$\varphi_j = \frac{1 + \sum_{i=1}^{n} 1\{z^{(i)} = j\}}{k + n}$$

(Note $\sum \varphi_j = 1$)

Thus, applying this for Naive Bayes:

$$\varphi_{j|y=1} = \frac{1 + \sum_{i=1}^{n} 1\{x_j^{(i)} = 1 \wedge y^{(i)} = 1\}}{2 + \sum_{i=1}^{n} 1\{y^{(i)} = 1\}}$$

$$\varphi_{j|y=0} = \frac{1 + \sum_{i=1}^{n} 1\{x_j^{(i)} = 1 \wedge y^{(i)} = 0\}}{2 + \sum_{i=1}^{n} 1\{y^{(i)} = 0\}}$$