USC Viterbi School of Engineering

ESTIMATION AND REGRESSION

EE 541 – UNIT 3B

DR. BRANDON FRANZKE

Spring 2023

REGRESSION OVERVIEW

- Regression is data fitting to a specific parameterized function class
- Linear regression
  - Same as LMMSE, but with data averages replacing expectation (ensemble averages)
    - Linear least-squares
  - Generalize on-line learning to full-batch and mini-batches
- Regularization *(later)*
- Logistical Regression *(later)*

GENERAL REGRESSION PROBLEM

Given a data set: $\mathcal{D} = \{(\mathbf{x}_n, \mathbf{y}_n)\}_{n=1}^{N}$

General regression problem:

$$\min_{\Theta} \left\langle C\big(\mathbf{y}, \mathbf{g}(\mathbf{x};\Theta)\big)\right\rangle_{\mathcal{D}} \qquad \Theta_{opt} = \arg\min_{\Theta} \left\langle C\big(\mathbf{y}, \mathbf{g}(\mathbf{x};\Theta)\big)\right\rangle_{\mathcal{D}} \qquad \hat{\mathbf{y}} = \mathbf{g}\big(\mathbf{x};\Theta_{\text{opt}}\big)$$

Empirical expectation (average over data):

$$\langle \mathbf{h}(\mathbf{x},\mathbf{y})\rangle_{\mathcal{S}} \equiv \frac{1}{|\mathcal{S}|} \sum_{(\mathbf{x}_n,\mathbf{y}_n)\in\mathcal{S}} \mathbf{h}(\mathbf{x}_n,\mathbf{y}_n)$$

x ~ *regressor* (*observed*)
y ~ target (*desired*)

For large averaging sets (i.e., many realizations):

$$\mathbb{E}\{\mathbf{h}\big(\mathbf{x}(t),\mathbf{y}(t)\big)\} = \int \mathbf{h}(\mathbf{x},\mathbf{y})\, p_{x(t),y(t)}(\mathbf{x},\mathbf{y})\, d\mathbf{y}\, d\mathbf{x} \approx \langle \mathbf{h}(\mathbf{x},\mathbf{y})\rangle_{\mathcal{S}}$$

sample mean

Monte Carlo method

LEAST-SQUARES (LS) REGRESSION PROBLEM

$$\min_{\Theta} \langle \|\mathbf{y} - \mathbf{g}(\mathbf{x};\Theta)\| \rangle_{\mathcal{D}} \quad \Leftrightarrow \quad \min_{\Theta} \sum_{n=1}^{N} \|\mathbf{y}_n - \mathbf{g}(\mathbf{x}_n;\Theta)\|^2$$

$$\Theta_{\text{opt}} = \arg\min_{\Theta} \langle \|\mathbf{y} - \mathbf{g}(\mathbf{x};\Theta)\|^2 \rangle_{\mathcal{D}}$$

Squared-error is a common cost function in (electrical) engineering

corresponds to **power or energy** in many applications

LINEAR AND AFFINE LEAST SQUARES REGRESSION

Linear regression problem:

$$\min_{\mathbf{W}} \langle \|\mathbf{y} - \mathbf{W}\mathbf{x}\|^2 \rangle_{\mathcal{D}} \quad \Leftrightarrow \quad \min_{\mathbf{W}} \sum_{n=1}^{N} \|\mathbf{y}_n - \mathbf{W}\mathbf{x}_n\|^2$$

$$\mathbf{W}_{\mathrm{LLSE}} = \arg\min_{\mathbf{W}} \langle \|\mathbf{y} - \mathbf{W}\mathbf{x}\|^2 \rangle_{\mathcal{D}}$$

$$\hat{\mathbf{y}} = \mathbf{W}_{\mathrm{LLSE}}\mathbf{x}$$

Affine regression (*a.k.a.*, Linear regression):

$$\mathbf{W}_{\mathrm{ALSE}}, \mathbf{b}_{\mathrm{ALSE}} = \arg\min_{\mathbf{W},\mathbf{b}} \langle \|\mathbf{y} - [\mathbf{W}\mathbf{x} + \mathbf{b}]\|^2 \rangle_{\mathcal{D}}$$

$$\hat{\mathbf{y}} = \mathbf{W}_{\mathrm{ALSE}}\mathbf{x} + \mathbf{b}_{\mathrm{ALSE}}$$

USC Viterbi School of Engineering

LINEAR AND AFFINE REGRESSION SOLUTION

Data averaging operator has linearity property like expectation

$$\mathbb{E}\{\mathrm{L}(x(t))\} = \mathrm{L}\left(\mathbb{E}(x(t))\right) \qquad \langle \mathrm{L}(x)\rangle = \mathrm{L}(\langle x \rangle)$$

This means the solutions are the same as the MMSE solutions with expectation replaced by data average

For example, Linear LS regression:

$$\mathbf{W}_{\mathrm{LLSE}} = \widehat{\mathbf{R}}_{\mathbf{YX}}\widehat{\mathbf{R}}_{\mathbf{X}}^{-1}$$

$$\hat{\mathbf{y}} = \widehat{\mathbf{R}}_{\mathbf{YX}}\widehat{\mathbf{R}}_{\mathbf{X}}^{-1}\mathbf{x}$$

$$\begin{aligned} LLS\varepsilon &= \langle \|\boldsymbol{y} - \mathbf{W}_{\mathrm{LLSE}}\mathbf{x}\|^2 \rangle \\ &= \langle \|\mathbf{y}\|^2 - \|\mathbf{W}_{\mathrm{LLSE}}\mathbf{x}\|^2 \rangle_{\mathcal{D}} \\ &= \mathrm{Tr}\left(\widehat{\mathbf{R}}_{\mathbf{Y}} - \widehat{\mathbf{R}}_{\mathbf{YX}}\widehat{\mathbf{R}}_{\mathbf{X}}^{-1}\,\widehat{\mathbf{R}}_{\mathbf{XY}}\right) \end{aligned}$$

$$\widehat{\mathbf{R}}_{\mathbf{X}} = \langle \mathbf{x}\mathbf{x}^T \rangle_{\mathcal{D}} = \frac{1}{N}\sum_{n=1}^{N} \mathbf{x}_n\mathbf{x}_n^T \qquad \widehat{\mathbf{R}}_{\mathbf{XY}} = \langle \mathbf{x}\mathbf{y}^T \rangle_{\mathcal{D}} = \frac{1}{N}\sum_{n=1}^{N} \mathbf{x}_n\mathbf{y}_n^T$$

PROOF FOR LLSE REGRESSION

$$\min_{\mathbf{W}} \langle \|\mathbf{y} - \mathbf{W}\mathbf{x}\|^2 \rangle_{\mathcal{D}}$$

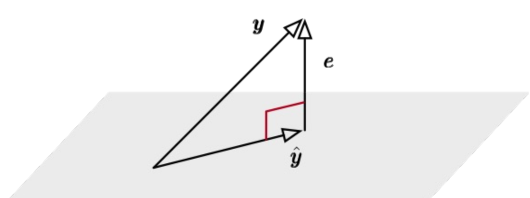
Proof:

$$\begin{aligned}
\mathrm{LSE}(\mathbf{G}) &= \langle \|\mathbf{y} - \mathbf{G}\mathbf{x}\|^2 \rangle \\
&= \left\langle \left\| (\mathbf{y} - \mathbf{G}_{\mathrm{opt}}\mathbf{x}) + (\mathbf{G}_{\mathrm{opt}} - \mathbf{G})\mathbf{x} \right\|^2 \right\rangle \\
&= \left\langle \left\| (\mathbf{y} - \mathbf{G}_{\mathrm{opt}}\mathbf{x}) \right\|^2 \right\rangle + \mathrm{Tr}\left((\mathbf{G}_{\mathrm{opt}} - \mathbf{G}) \widehat{\mathbf{R}}_{\mathbf{X}} (\mathbf{G}_{\mathrm{opt}} - \mathbf{G})^T \right) \\
&\quad + 2\,\mathrm{Tr}\left((\widehat{\mathbf{R}}_{\mathbf{YX}} - \mathbf{G}_{\mathrm{opt}} \widehat{\mathbf{R}}_{\mathbf{X}}) (\mathbf{G}_{\mathrm{opt}} - \mathbf{G})^{\mathrm{T}} \right)
\end{aligned}$$

$\mathrm{v}^{\mathrm{T}}\mathrm{w} = \mathrm{Tr}(\mathrm{w}\mathrm{v}^{\mathrm{T}})$

if: $\mathbf{G}_{\mathrm{opt}} \widehat{\mathbf{R}}_{\mathbf{X}} = \widehat{\mathbf{R}}_{\mathbf{XY}}$ then: $\mathrm{LSE}(\mathbf{G}) = \mathbb{E}\left\{ \left\| \mathbf{y} - \mathbf{G}_{\mathrm{opt}} x \right\|^2 \right\} + \underbrace{\mathrm{Tr}\left((\mathbf{G}_{\mathrm{opt}} - \mathbf{G}) \widehat{\mathbf{R}}_{\mathbf{X}} (\mathbf{G}_{\mathrm{opt}} - \mathbf{G})^{\mathrm{T}} \right)}_{\geq 0\ \forall \mathbf{G},\ \text{since } \widehat{\mathbf{R}}_{\mathbf{X}} \text{ is psd}}$

Wiener-Hopf equations *(Orthogonality Principle)*



space of all estimates/approximations

because of orthogonality principle
(error and signal uncorrelated)

$$\begin{aligned}
\langle \|\mathbf{y} - \hat{\mathbf{y}}\|^2 \rangle &= \langle \|\mathbf{y}\|^2 \rangle + \langle \|\hat{\mathbf{y}}\|^2 \rangle \\
&= \mathrm{Tr}\left(\widehat{\mathbf{R}}_{\mathbf{Y}} - \widehat{\mathbf{R}}_{\mathbf{YX}} \widehat{\mathbf{R}}_{\mathbf{X}}^{-1} \widehat{\mathbf{R}}_{\mathbf{XY}} \right)
\end{aligned}$$

SOLUTION TO LINEAR AND AFFINE (LS) REGRESSION

It makes intuitive sense:

For LMMSE estimate: if you did not know the second moments you would estimate these correlations from data

*in addition to optimality in the Gaussian case, linear MMSE estimation is popular because it **requires much less data to accurately estimate second moments** than a complete statistical description (or higher moments)*

LLSE REGRESSION: SCALAR FROM SCALAR

Estimate y from x

Linear regression problem:

$$\min_w \langle (y - wx)^2 \rangle \quad \Leftrightarrow \quad \min_w \frac{1}{N} \sum_{n=1}^{N} (y_n - wx_n)^2$$

Solution (special case):

$$w_{LLSE} = \frac{\hat{r}_{yx}}{\hat{r}_x}$$

$$\hat{y} = \frac{\hat{r}_{yx}}{\hat{r}_x} x$$

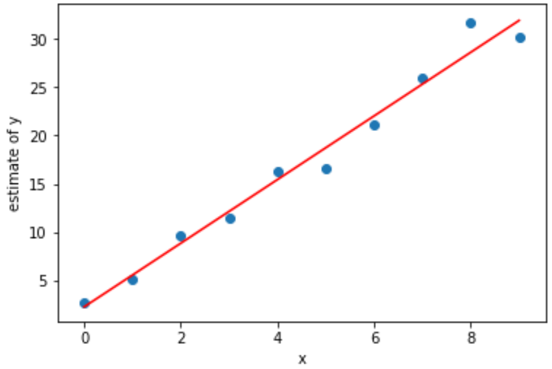
$$\hat{r}_x = \langle x^2 \rangle = \frac{1}{N} \sum_{n=1}^{N} x_n^2$$

$$\hat{r}_{yx} = \langle yx \rangle = \frac{1}{N} \sum_{n=1}^{N} y_n x_n$$

$$\begin{aligned} LLS\varepsilon &= \langle [y - w_{LLSE} x]^2 \rangle \\ &= \langle y^2 \rangle - \langle [w_{LLSE} x]^2 \rangle \\ &= \hat{r}_y - \hat{r}_{yx}^2 \hat{r}_x^{-1} \end{aligned}$$

if sample means all 0:

$$= \hat{\sigma}_y^2 (1 - \hat{\rho}^2)$$



LLSE REGRESSION: SCALAR FROM SCALAR

Estimate y from x

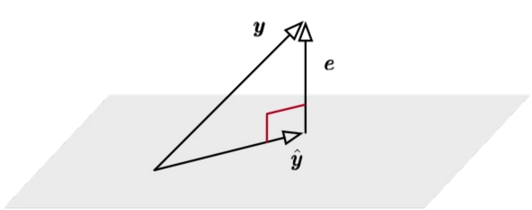
Linear regression problem:

$$\min_w \langle (y - wx)^2 \rangle \quad \Leftrightarrow \quad \min_w \frac{1}{N}\sum_{n=1}^{N}(y_n - wx_n)^2 \quad \Leftrightarrow \quad \|\mathbf{y} - w\mathbf{x}\|^2$$

Solution (special case):

$$w_{LLSE} = \frac{\mathbf{y}^T\mathbf{x}}{\mathbf{x}^T\mathbf{x}} \qquad (N)LLS\varepsilon = \|\mathbf{y}\|^2 - \left(\frac{\mathbf{y}^T\mathbf{x}}{\mathbf{x}^T\mathbf{x}}\right)^2 \|\mathbf{x}\|^2 \qquad y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} \qquad x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix}$$

$$\hat{\mathbf{y}} = \frac{\mathbf{y}^T\mathbf{x}}{\mathbf{x}^T\mathbf{x}}\mathbf{x} \qquad = \|\mathbf{y}\|^2 - \frac{(\mathbf{y}^T\mathbf{x})^2}{\|\mathbf{x}\|^2}$$



space of all estimates/approximations

this "stacked" approach yields the same as the $\langle\cdot\rangle_D$ approach on the previous slides

$\hat{y}$ stacked in a vector

LLSE REGRESSION: SCALAR FROM VECTOR

Estimate y from x

Linear regression problem:

$$\min_{\mathbf{w}} \langle (y - \mathbf{w}^T \mathbf{x})^2 \rangle \quad \Leftrightarrow \quad \min_{\mathbf{w}} \frac{1}{N} \sum_{n=1}^{N} (y_n - \mathbf{w}^T \mathbf{x}_n)^2$$

$$\hat{y} = \mathbf{w}^T \mathbf{x}$$

$$w = \widehat{\mathbf{R}}_{\mathrm{X}}^{-1} \hat{\mathbf{r}}_{\mathbf{x}y}$$

$$\hat{\mathbf{r}}_{\mathbf{x}y} = \widehat{\mathbf{R}}_{\mathbf{xy}} = \langle \mathbf{x} y \rangle$$

$$LLS\varepsilon = \hat{r}_{\mathrm{y}} - \hat{\mathbf{r}}_{\mathbf{x}y}^T \widehat{\mathbf{R}}_{\mathrm{X}}^{-1} \hat{\mathbf{r}}_{\mathbf{x}y}$$

$$\widehat{\mathbf{R}}_{\mathbf{x}} \mathbf{w} = \hat{\mathbf{r}}_{\mathbf{x}y}$$

$$\widehat{\mathbf{R}}_{\mathrm{X}} \mathbf{w} = \hat{\mathbf{r}}_{\mathbf{x}y}$$ *"Normal Equations"*

similar: just change $\mathbb{E}[\cdot]$ to $\langle \cdot \rangle_D$ in LMMSE result

what about the "stacked" approach for this case??

LLSE REGRESSION: SCALAR FROM VECTOR

Estimate y from x

Linear regression problem:

$$\min_w \langle (y - w^T x)^2 \rangle \quad \Leftrightarrow \quad \min_w \frac{1}{N} \sum_{n=1}^{N} (y_n - w^T x)^2 \quad \Leftrightarrow \quad \min_w \|y - Xw\|^2$$

Solution (special case):

$$\hat{\mathbf{y}} = \boldsymbol{X}\boldsymbol{w}$$

$$\mathbf{w} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$$

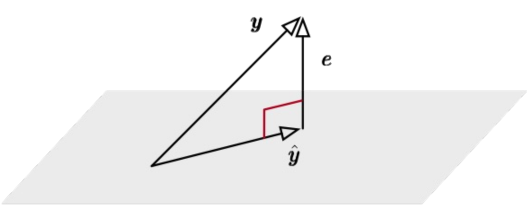
$$\hat{\mathbf{y}} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} = \mathbf{P}_\mathbf{X}\mathbf{y}$$

$$(N)LLS\varepsilon = \mathrm{Tr}(\|\mathbf{y}\|^2 - \|\mathbf{P}_\mathbf{x}\mathbf{y}\|^2) = \mathrm{Tr}(\|(\mathbf{I} - \mathbf{P}_\mathbf{x})\mathbf{y}\|^2)$$

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1^T \\ \mathbf{x}_2^T \\ \vdots \\ \mathbf{x}_N^T \end{bmatrix} \qquad \mathbf{X}^T = [\mathbf{x}_1 \quad \mathbf{x}_2 \quad \cdots \quad \mathbf{x}_N]$$

$$\mathbf{X}^T\mathbf{X} = \mathbf{X}^T\mathbf{y}$$

normal equations



space of all estimates/approximations

this is the same as $\langle\cdot\rangle_D$ case, with all $\hat{y}$ stacked in a vector

$$\hat{\mathbf{R}}_\mathbf{X}^{-1}\hat{\mathbf{r}}_{\mathbf{x}y} = \left(\frac{1}{N}\mathbf{X}^T\mathbf{X}\right)^{-1}\left[\frac{1}{N}\mathbf{X}^T\mathbf{y}\right]$$

THE AFFINE TO LINEAR MATH "TRICK"

$$\min_{\mathbf{w}} \langle (y - \mathbf{w}^T\mathbf{x})^2 \rangle \quad \Leftrightarrow \quad \min_{\mathbf{w}} \frac{1}{N}\sum_{n=1}^{N}(y_n - \mathbf{w}^T\mathbf{x}_n)^2$$

$$\begin{aligned}\hat{y} &= [\mathbf{x}^T \quad | \quad 1]\begin{bmatrix}\mathbf{w}\\ b\end{bmatrix}\\ &= \mathbf{x}^T\mathbf{w} + b\end{aligned}$$

$$\begin{aligned}\hat{\mathbf{y}} &= [\mathbf{X} \quad | \quad \mathbf{1}]\begin{bmatrix}\mathbf{w}\\ b\end{bmatrix}\\ &= \mathbf{X}\mathbf{w} + \mathbf{b}1\end{aligned}$$

$$[\mathbf{X} \quad | \quad \mathbf{1}] = \begin{bmatrix}\mathbf{x}_1^T & | & 1\\ \mathbf{x}_2^T & | & 1\\ \vdots & | & \vdots\\ \mathbf{x}_N^T & | & 1\end{bmatrix}$$

therefore: compact notation <u>even if using bias</u> (b) term

LINEAR CLASSIFICATION

ESTIMATION, REGRESSION, CLASSIFICATION

statistical models

MMSE Estimation

Linear/Affine MMSE Est.

FIR Wiener filtering

GD, SGD, LMS

Bayesian decision theory

Hard decisions

soft decisions (APP)

ML/MAP parameter estimation

Karhunen-Loeve expansion

sufficient statistics

data driven

general regression

linear LS regression

stochastic gradient and

Classification from data

linear classifier

logistical regression (perceptron)

regularization

PCA

feature design

neural networks

for regression and classification

learning with SGD

working with data

LINEAR CLASSIFIER

perform linear regression and then **threshold** to hard decision

Example:

$$y \in \{-1, +1\}$$

$$\hat{y} = sign(\mathbf{w}^T\mathbf{x})$$

$$sign(v) = \begin{cases} +1, & v \geq 0 \\ -1, & v < 0 \end{cases}$$

$$\min_{\mathbf{w}} \frac{1}{N} \sum_{n=1}^{N} (y_n - \mathbf{w}^T\mathbf{x}_n)^2$$

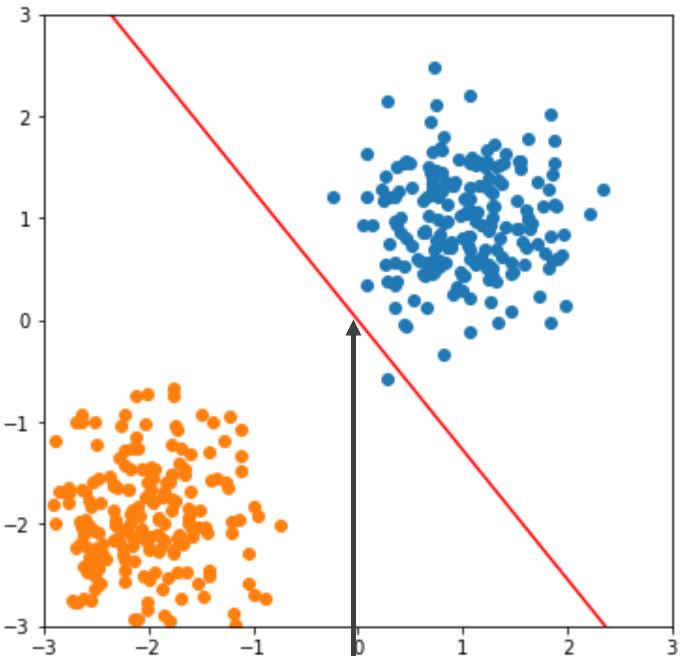
standard LLSE regression with prediction thresholding

Review: linear_classifier_examples.ipynb

EXAMPLE: LINEAR AND AFFINE REGRESSION

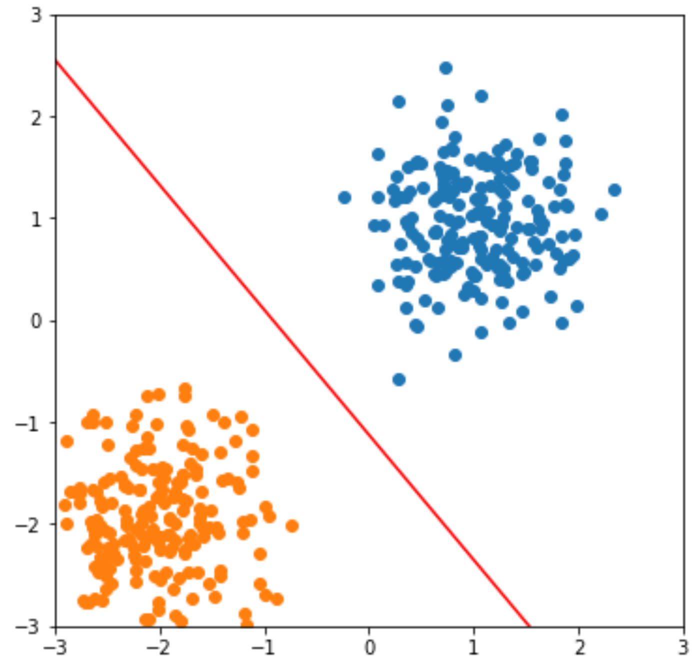
$$\hat{y} = \mathbf{x}^T \mathbf{w}$$

$$\hat{y} = [\mathbf{x}^T \quad | \quad 1] \begin{bmatrix} \mathbf{w} \\ b \end{bmatrix}$$



$$s_0 = \begin{bmatrix} +1 \\ +1 \end{bmatrix}$$

$$s_1 = \begin{bmatrix} -2 \\ -2 \end{bmatrix}$$



for the case with no bias term, the decision threshold has to pass through the origin

adding the bias term allows for offset from the origin

MAXIMUM LIKELIHOOD ESTIMATION EXAMPLE

this is a model for the data $\{(x_n, y_n)\}$:

$$y_n = \mathbf{w}^T \mathbf{x}_n + v_n, \qquad n = 1,2,\ldots,N$$

$$\mathbf{y} = \mathbf{X}\mathbf{w} + \mathbf{v}(t)$$

$$p_{v(t)}(\mathbf{v}) = \mathcal{N}_N(\mathbf{v}; \mathbf{0}, \sigma_v^2 \mathbf{I})$$

$$p_{\mathbf{y}(t)|\mathbf{X}(t)}(\mathbf{y}|\mathbf{X};\mathbf{w}) = p_{\mathbf{v}(t)}(\mathbf{y} - \mathbf{X}\mathbf{w}) = \mathcal{N}_N(\mathbf{v}; \mathbf{X}\mathbf{w}, \sigma_v^2 \mathbf{I})$$

$$NLL(\mathbf{w}) = -\ln\left(p_{\mathbf{y}(t)}(\mathbf{y}|\mathbf{X};\mathbf{w})\right)$$

$$= -\ln\left(\frac{1}{(2\pi\sigma_v^2)^{\frac{N}{2}}} \exp\left[-\frac{1}{2\sigma_v^2}\|\mathbf{y} - \mathbf{X}\mathbf{w}\|^2\right]\right)$$

$$= -\frac{1}{2\sigma_v^2}\|\mathbf{y} - \mathbf{X}\mathbf{w}\|^2 + \frac{\mathrm{N}}{2}\ln(2\pi\sigma_v^2)$$

$$\boxed{\max_{\mathbf{w}} p_{\mathbf{y}(t)|\mathbf{X}(t)}(\mathbf{y}|\mathbf{X};\mathbf{w}) \Leftrightarrow \min_{\mathbf{w}} \|\mathbf{y} - \mathbf{X}\mathbf{w}\|^2}$$

Maximum Likelihood <==> Minimize Neg-Log-Likelihood <==> LLSE regression

(under **this** model for the data)

PROPERTIES OF ML ESTIMATORS

- Asymptotically Gaussian:
  - For large amounts of data, the ML estimate is Gaussian with mean equal to the true parameter (models matched)
- Consistent:
  - The limit in probability of the ML estimate is the true parameter (model matched)
- The ML estimate minimizes the KL Divergence between the model distribution and the empirical data distribution. KL divergence measures the difference between two distribution (Info. Theory).
  - Minimizing KL divergence in this case also corresponds to minimizing the cross entropy

INFORMATION THEORY

ML ESTIMATION AND INFORMATION THEORY

Entropy:

$$H\big(X(t)\big) = \mathbb{E}\left\{\log_2\left(\frac{1}{p_{X(t)}\big(X(t)\big)}\right)\right\}$$
$$= \sum_k p_{X(t)}(k)\log_2\left(\frac{1}{p_{X(t)}(k)}\right)$$
$$= \sum_k p_k \log_2\left(\frac{1}{p_k}\right)$$

Intuition:

events with low probability have large information — *e.g.*, "it will snow in Phoenix tomorrow"

the entropy is the average information learned when the value of $X(u)$ is revealed.

Examples:

weather report in Phoenix has low entropy (almost always the same), whereas in Sioux City, SD it has high entropy (highly variant weather)

fair die:

$$H(X(t)) = \log_2(1/6) = 2.58\ bits/roll$$

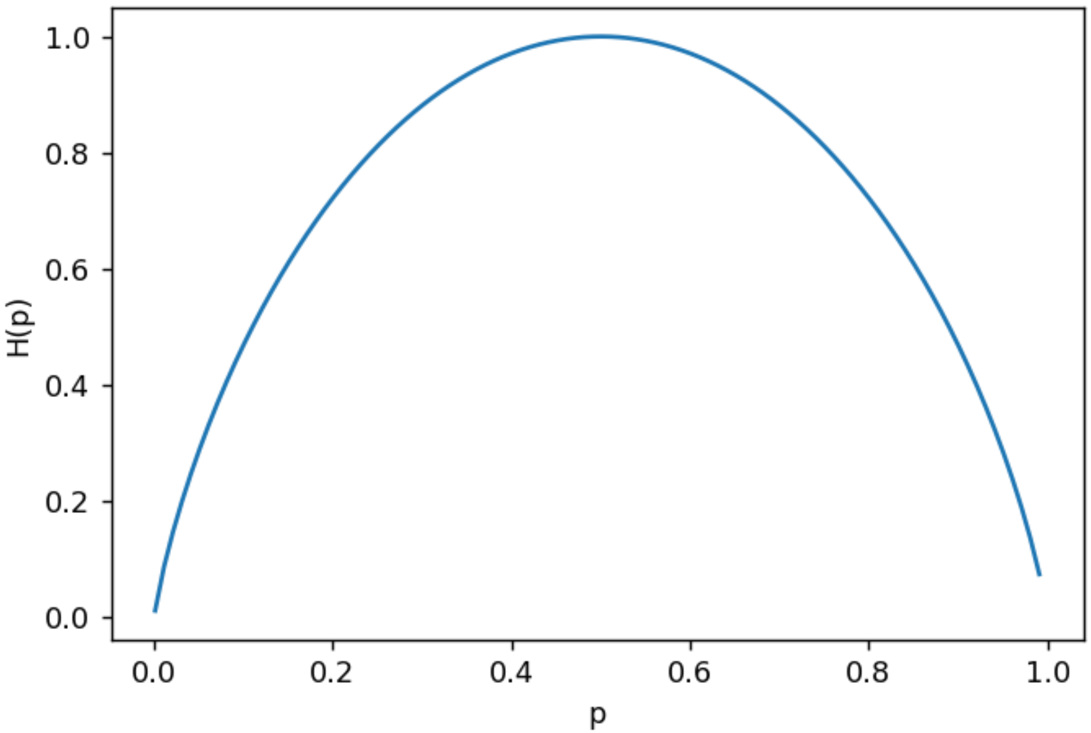
loaded die:

$$\begin{aligned} H\big(X(t)\big) &= -0.4\log_2(0.4) - 0.1\log_2(0.1) - 0.01\log_2(0.01) \\ &\quad - 0.09\log_2(0.09) - 0.25\log_2(0.25) - 0.15\log_2(0.15) \\ &= 2.15\ bits/roll \end{aligned}$$

ML ESTIMATION AND INFORMATION THEORY

Entropy of i.i.d. Bernoulli Source (with success probability p)

$$H(p) = -p\log_2(p) - (1-p)\log_2(1-p)$$



ML ESTIMATION AND INFORMATION THEORY

KL-Divergence

$$D(p \parallel \tilde{p}) = \mathbb{E}_p\left\{\log\left(\frac{p_x(X(t))}{\tilde{p}_x(X(t))}\right)\right\}$$

$$= \sum_k p_k \log\left(\frac{p_k}{\tilde{p}_k}\right)$$

$$= \sum_k p_k \log(p_k) - \sum_k p_k \log(\tilde{p}_k)$$

$$= CE(p, \tilde{p}) - H(p)$$

Cross-Entropy

$$CE(p, \tilde{p}) = \mathbb{E}_p\left\{\log\left(\frac{1}{\tilde{p}(X(t))}\right)\right\}$$

ML ESTIMATION AND INFORMATION THEORY

ML parameter estimation minimizes empirical CE (and KL divergence)

$$p_{data}(y|\mathbf{x}) = \text{data distribution of the data (typically unknown)}$$

$$p_{model}(y|\mathbf{x};\Theta) = \text{modeled distribution of the data (function of parameters)}$$

$$\begin{aligned} CE(p_{data}, p_{model}(\Theta)) &= \mathbb{E}_{p_{data}(y|x)}\left\{\log\left(\frac{1}{p_{model}(y(t)|\mathbf{x}(t);\Theta)}\right)\right\} \\ &\approx \left\langle -\log\left(p_{model}(y|\mathbf{x};\Theta)\right)\right\rangle_{\mathcal{D}} \\ &= \frac{1}{N}\sum_{n=1}^{N}\log\left(p_{model}(y_n|\mathbf{x}_n;\Theta)\right) \end{aligned}$$

$$\max_{\Theta} p_{model}(\mathbf{y}|\mathbf{X};\Theta) \quad \Leftrightarrow \quad \min_{\theta}(-\log(p_{model}(\mathbf{y}|\mathbf{X};\Theta)))$$

$$\Leftrightarrow \quad \min_{\theta}\left(-\sum_{n=1}^{N}\log\left(p_{model}(y_n|\mathbf{x}_n;\Theta)\right)\right)$$

(i.i.d. y_n assumed)

$$\Leftrightarrow \quad \min_{\theta}\left(-\frac{1}{N}\sum_{n=1}^{N}\log\left(p_{model}(y_n|\mathbf{x}_n;\Theta)\right)\right)$$

(empirical Cross-Entropy)

Max-Likelihood Estimation of neural network weights is always minimizing the empirical cross entropy between data distribution and the modeled distribution

MULTI-CLASS CROSS ENTROPY EXAMPLE

One hot encoding:

cat: 0
dog: 1
bird: 2

Sample data labels:

n=1: y=1 (dog)
n=2: y=2 (bird)
n=3: y=0 (cat)

Classifier Output:

n=1: [0.3, 0.5, 0.2]
n=2: [0, 0, 1]
n=3: [0.4, 0.5, 0.1]

[p(cat), p(dog), p(bird)]

$$Loss = -\frac{1}{3}[\log(0.5) + \log(1) + \log(0.4)]$$

$$\overline{MCE}\big(p_{data}, p_{model}(w)\big) = -\frac{1}{N}\sum_{n=1}^{N}\sum_{m=1}^{M}\mathbb{I}[y_n = m]\log\big(p_{model}(y_n = m; w)\big)$$