CEE 697M: Big Data and Machine Learning for Engineers
Lecture 1d: Decision and Information Theories

Jimi Oke

UMass Amherst

College of Engineering

September 11, 2025

Outline

Basics

The posterior expected loss/risk for an action a given a state of nature h is:

Basics

The posterior expected loss/risk for an action a given a state of nature h is:

$$R(a|\boldsymbol{x}) := \mathbb{E}_{p(h|\boldsymbol{x})}[\ell(h,a) = \sum_{h\in\mathcal{H}} \ell(h,a)p(h|\boldsymbol{x}) \tag{1}$$

Basics

The posterior expected loss/risk for an action a given a state of nature h is:

$$R(a|\boldsymbol{x}) := \mathbb{E}_{p(h|\boldsymbol{x})}[\ell(h, a) = \sum_{h \in \mathcal{H}} \ell(h, a) p(h|\boldsymbol{x}) \tag{1}$$

In making decisions, we want to find an optimal policy π^* by minimizing risk:

Basics

The posterior expected loss/risk for an action a given a state of nature h is:

$$R(a|\boldsymbol{x}) := \mathbb{E}_{p(h|\boldsymbol{x})}[\ell(h, a) = \sum_{h \in \mathcal{H}} \ell(h, a) p(h|\boldsymbol{x}) \tag{1}$$

In making decisions, we want to find an optimal policy π^* by minimizing risk:

$$\pi^*(\boldsymbol{x}) = \arg\min_{a \in \mathcal{A}} \mathbb{E}_{p(h|\boldsymbol{x}}[\ell(h, a)] \tag{2}$$

Basics

The posterior expected loss/risk for an action a given a state of nature h is:

$$R(a|\boldsymbol{x}) := \mathbb{E}_{p(h|\boldsymbol{x})}[\ell(h, a) = \sum_{h \in \mathcal{H}} \ell(h, a) p(h|\boldsymbol{x}) \tag{1}$$

In making decisions, we want to find an optimal policy π^* by minimizing risk:

$$\pi^*(\boldsymbol{x}) = \arg\min_{a \in \mathcal{A}} \mathbb{E}_{p(h|\boldsymbol{x}}[\ell(h, a)] \tag{2}$$

or maximizing expected utility $\mathbb{U}(h, a) = -\ell(h, a)$:

$$\pi^*(\boldsymbol{x}) = \arg\max_{a \in \mathcal{A}} \mathbb{E}_h[U(h, a)] \tag{3}$$

Classification problems

Classification problems

To assign the optimal class label in a classification prediction, the **optimal policy** is:

Classification problems

To assign the optimal class label in a classification prediction, the **optimal policy** is:

$$\pi^*(\boldsymbol{x}) = \arg\max_{y \in \mathcal{Y}} p(y|\boldsymbol{x}) \tag{4}$$

Classification problems

To assign the optimal class label in a classification prediction, the **optimal policy** is:

$$\pi^*(\boldsymbol{x}) = \arg\max_{y \in \mathcal{Y}} p(y|\boldsymbol{x}) \tag{4}$$

that is, we assign the label to class that is most probable.

Classification problems

To assign the optimal class label in a classification prediction, the **optimal policy** is:

$$\pi^*(\boldsymbol{x}) = \arg\max_{y \in \mathcal{Y}} p(y|\boldsymbol{x}) \tag{4}$$

that is, we assign the label to class that is most probable.

- $y \in \{0, 1\}$: true label
- $\hat{y} \in \{0, 1\}$: predicted label
- $\boldsymbol{x}$: input vector

Classification problems

To assign the optimal class label in a classification prediction, the **optimal policy** is:

$$\pi^*(\boldsymbol{x}) = \arg\max_{y \in \mathcal{Y}} p(y|\boldsymbol{x}) \tag{4}$$

that is, we assign the label to class that is most probable.

- $y \in \{0, 1\}$: true label
- $\hat{y} \in \{0, 1\}$: predicted label
- $\boldsymbol{x}$: input vector

The posterior expected loss (if the loss function is the 0-1 loss) is:

Classification problems

To assign the optimal class label in a classification prediction, the **optimal policy** is:

$$\pi^*(\boldsymbol{x}) = \arg\max_{y \in \mathcal{Y}} p(y|\boldsymbol{x}) \tag{4}$$

that is, we assign the label to class that is most probable.

- $y \in \{0, 1\}$: true label
- $\hat{y} \in \{0, 1\}$: predicted label
- $\boldsymbol{x}$: input vector

The posterior expected loss (if the loss function is the 0-1 loss) is:

$$R(\hat{y}|\boldsymbol{x}) = p(\hat{y} \neq y^*|\boldsymbol{x}) \tag{5}$$

Classification problems

To assign the optimal class label in a classification prediction, the **optimal policy** is:

$$\pi^*(\boldsymbol{x}) = \arg\max_{y \in \mathcal{Y}} p(y|\boldsymbol{x}) \tag{4}$$

that is, we assign the label to class that is most probable.

- $y \in \{0, 1\}$: true label
- $\hat{y} \in \{0, 1\}$: predicted label
- $\boldsymbol{x}$: input vector

The posterior expected loss (if the loss function is the 0-1 loss) is:

$$R(\hat{y}|\boldsymbol{x}) = p(\hat{y} \neq y^*|\boldsymbol{x}) \tag{5}$$

(This is the error rate)

Decision rule for binary classification

Given a probability threshold τ, we can assign a class label in a binary setting using:

Decision rule for binary classification

Given a probability threshold τ, we can assign a class label in a binary setting using:

$$\hat{y}(\boldsymbol{x}) = \mathbb{I}(p(y = 1|\boldsymbol{x}) \geq 1 - \tau) \tag{6}$$

Summarizing performance

Summarizing performance

- Precision:

Summarizing performance

- Precision:

$$\mathcal{P}(\tau) :=$$

Summarizing performance

- Precision:

$$\mathcal{P}(\tau) := p(y = 1|\hat{y} = 1, \tau)$$

Summarizing performance

- Precision:

$$\mathcal{P}(\tau) := p(y = 1|\hat{y} = 1, \tau) = \frac{TP_\tau}{TP_\tau + FP_\tau} \tag{7}$$

Summarizing performance

- Precision:

$$\mathcal{P}(\tau) := p(y = 1|\hat{y} = 1, \tau) = \frac{TP_\tau}{TP_\tau + FP_\tau} \tag{7}$$

- Recall (sensitivity, hit rate, true positive rate (TPR):

Summarizing performance

- Precision:

$$\mathcal{P}(\tau) := p(y = 1|\hat{y} = 1, \tau) = \frac{TP_\tau}{TP_\tau + FP_\tau} \tag{7}$$

- Recall (sensitivity, hit rate, true positive rate (TPR):

$$\mathcal{R}(\tau) :=$$

Summarizing performance

- Precision:

$$\mathcal{P}(\tau) := p(y = 1|\hat{y} = 1, \tau) = \frac{TP_\tau}{TP_\tau + FP_\tau} \tag{7}$$

- Recall (sensitivity, hit rate, true positive rate (TPR):

$$\mathcal{R}(\tau) := p(\hat{y} = 1|y = 1, \tau)$$

Summarizing performance

- Precision:

$$\mathcal{P}(\tau) := p(y = 1|\hat{y} = 1, \tau) = \frac{TP_\tau}{TP_\tau + FP_\tau} \tag{7}$$

- Recall (sensitivity, hit rate, true positive rate (TPR):

$$\mathcal{R}(\tau) := p(\hat{y} = 1|y = 1, \tau) = \frac{TP_\tau}{TP_\tau + FN_\tau} \tag{8}$$

Summarizing performance

- Precision:

$$\mathcal{P}(\tau) := p(y = 1|\hat{y} = 1, \tau) = \frac{TP_\tau}{TP_\tau + FP_\tau} \tag{7}$$

- Recall (sensitivity, hit rate, true positive rate (TPR):

$$\mathcal{R}(\tau) := p(\hat{y} = 1|y = 1, \tau) = \frac{TP_\tau}{TP_\tau + FN_\tau} \tag{8}$$

- False positive rate (FPR, false alarm rate, type I error rate):

Summarizing performance

- Precision:

$$\mathcal{P}(\tau) := p(y = 1|\hat{y} = 1, \tau) = \frac{TP_\tau}{TP_\tau + FP_\tau} \quad (7)$$

- Recall (sensitivity, hit rate, true positive rate (TPR):

$$\mathcal{R}(\tau) := p(\hat{y} = 1|y = 1, \tau) = \frac{TP_\tau}{TP_\tau + FN_\tau} \quad (8)$$

- False positive rate (FPR, false alarm rate, type I error rate):

$$FPR(\tau) = p(\hat{y} = 1|y = 0, \tau)$$

Summarizing performance

- Precision:

$$\mathcal{P}(\tau) := p(y = 1|\hat{y} = 1, \tau) = \frac{TP_\tau}{TP_\tau + FP_\tau} \tag{7}$$

- Recall (sensitivity, hit rate, true positive rate (TPR):

$$\mathcal{R}(\tau) := p(\hat{y} = 1|y = 1, \tau) = \frac{TP_\tau}{TP_\tau + FN_\tau} \tag{8}$$

- False positive rate (FPR, false alarm rate, type I error rate):

$$FPR(\tau) = p(\hat{y} = 1|y = 0, \tau) = \frac{FP_\tau}{FP_\tau + TN_\tau}$$

Summarizing performance

- Precision:

$$\mathcal{P}(\tau) := p(y = 1|\hat{y} = 1, \tau) = \frac{TP_\tau}{TP_\tau + FP_\tau} \tag{7}$$

- Recall (sensitivity, hit rate, true positive rate (TPR):

$$\mathcal{R}(\tau) := p(\hat{y} = 1|y = 1, \tau) = \frac{TP_\tau}{TP_\tau + FN_\tau} \tag{8}$$

- False positive rate (FPR, false alarm rate, type I error rate):

$$FPR(\tau) = p(\hat{y} = 1|y = 0, \tau) = \frac{FP_\tau}{FP_\tau + TN_\tau} = \frac{FP_\tau}{N} \tag{9}$$

Summarizing performance

- Precision:

$$\mathcal{P}(\tau) := p(y = 1|\hat{y} = 1, \tau) = \frac{TP_\tau}{TP_\tau + FP_\tau} \tag{7}$$

- Recall (sensitivity, hit rate, true positive rate (TPR):

$$\mathcal{R}(\tau) := p(\hat{y} = 1|y = 1, \tau) = \frac{TP_\tau}{TP_\tau + FN_\tau} \tag{8}$$

- False positive rate (FPR, false alarm rate, type I error rate):

$$FPR(\tau) = p(\hat{y} = 1|y = 0, \tau) = \frac{FP_\tau}{FP_\tau + TN_\tau} = \frac{FP_\tau}{N} \tag{9}$$

- F_β-score:

Summarizing performance

- Precision:

$$\mathcal{P}(\tau) := p(y = 1|\hat{y} = 1, \tau) = \frac{TP_\tau}{TP_\tau + FP_\tau} \tag{7}$$

- Recall (sensitivity, hit rate, true positive rate (TPR):

$$\mathcal{R}(\tau) := p(\hat{y} = 1|y = 1, \tau) = \frac{TP_\tau}{TP_\tau + FN_\tau} \tag{8}$$

- False positive rate (FPR, false alarm rate, type I error rate):

$$FPR(\tau) = p(\hat{y} = 1|y = 0, \tau) = \frac{FP_\tau}{FP_\tau + TN_\tau} = \frac{FP_\tau}{N} \tag{9}$$

- F_β-score:

$$F_\beta :=$$

Summarizing performance

- Precision:

$$\mathcal{P}(\tau) := p(y = 1|\hat{y} = 1, \tau) = \frac{TP_\tau}{TP_\tau + FP_\tau} \tag{7}$$

- Recall (sensitivity, hit rate, true positive rate (TPR):

$$\mathcal{R}(\tau) := p(\hat{y} = 1|y = 1, \tau) = \frac{TP_\tau}{TP_\tau + FN_\tau} \tag{8}$$

- False positive rate (FPR, false alarm rate, type I error rate):

$$FPR(\tau) = p(\hat{y} = 1|y = 0, \tau) = \frac{FP_\tau}{FP_\tau + TN_\tau} = \frac{FP_\tau}{N} \tag{9}$$

- F_β-score:

$$F_\beta := (1 + \beta^2)\frac{\mathcal{P} \cdot \mathcal{R}}{\beta^2\mathcal{P} + \mathcal{R}} \tag{10}$$

Summarizing performance

- Precision:

$$\mathcal{P}(\tau) := p(y = 1|\hat{y} = 1, \tau) = \frac{TP_\tau}{TP_\tau + FP_\tau} \tag{7}$$

- Recall (sensitivity, hit rate, true positive rate (TPR):

$$\mathcal{R}(\tau) := p(\hat{y} = 1|y = 1, \tau) = \frac{TP_\tau}{TP_\tau + FN_\tau} \tag{8}$$

- False positive rate (FPR, false alarm rate, type I error rate):

$$FPR(\tau) = p(\hat{y} = 1|y = 0, \tau) = \frac{FP_\tau}{FP_\tau + TN_\tau} = \frac{FP_\tau}{N} \tag{9}$$

- F_β-score:

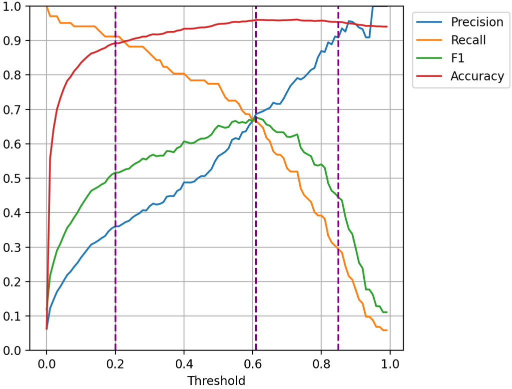
$$F_\beta := (1 + \beta^2)\frac{\mathcal{P} \cdot \mathcal{R}}{\beta^2 \mathcal{P} + \mathcal{R}} \tag{10}$$

Setting $\beta = 1$ gives the harmonic mean of precision and recall F_1.

Comparing performance measures to select threshold

Comparing performance measures to select threshold

Here we choose τ to maximize $\mathcal{P}$, $\mathcal{R}$ and F_1:

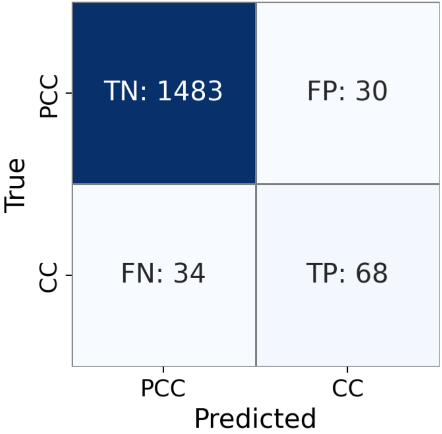


Activity: computing performance metrics

Given the confusion matrix, find $\mathcal{P}$, $\mathcal{R}$ and F_1:

Activity: computing performance metrics

Given the confusion matrix, find $\mathcal{P}$, $\mathcal{R}$ and F_1:



Performance curves

Performance curves

- ROC curves: TPR versus FPR for various τ

Performance curves

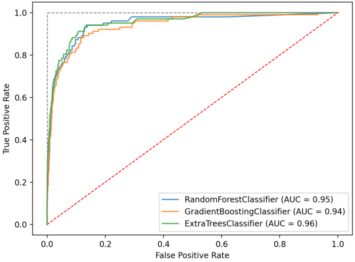
- ROC curves: TPR versus FPR for various τ
- Precision-recall curves: $\mathcal{P}$ versus $\mathcal{R}$ for various τ

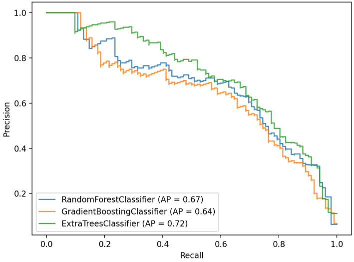
Performance curves

- ROC curves: TPR versus FPR for various τ
- Precision-recall curves: $\mathcal{P}$ versus $\mathcal{R}$ for various τ
- ROC and PRC of 3 candidate models:

Performance curves

- ROC curves: TPR versus FPR for various τ
- Precision-recall curves: $\mathcal{P}$ versus $\mathcal{R}$ for various τ
- ROC and PRC of 3 candidate models:





Regression problems

Regression problems

We find optimal parameters by minimizing loss functions such as:

Regression problems

We find optimal parameters by minimizing loss functions such as:

- L2 loss: squared error: $\ell_2(h, a) = (h - a)^2$

Regression problems

We find optimal parameters by minimizing loss functions such as:

- L2 loss: squared error: $\ell_2(h, a) = (h - a)^2$
- L1 loss: absolute value: $\ell_2(h, a) = |h - a|$

Regression problems

We find optimal parameters by minimizing loss functions such as:

- L2 loss: squared error: $\ell_2(h, a) = (h - a)^2$
- L1 loss: absolute value: $\ell_2(h, a) = |h - a|$(robust to outliers)

Regression problems

We find optimal parameters by minimizing loss functions such as:

- L2 loss: squared error: $\ell_2(h, a) = (h - a)^2$
- L1 loss: absolute value: $\ell_2(h, a) = |h - a|$(robust to outliers)
- Huber loss:

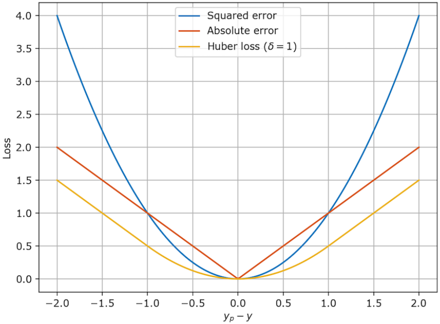
$$\ell_\delta(h, a) = \begin{cases} \frac{(h-a)^2}{2}, & |h - a| \leq \delta \\ \delta|h - a| - \delta^2/2, & |h - a| > \delta \end{cases} \tag{11}$$

Regression problems

We find optimal parameters by minimizing loss functions such as:

- L2 loss: squared error: $\ell_2(h, a) = (h - a)^2$
- L1 loss: absolute value: $\ell_2(h, a) = |h - a|$(robust to outliers)
- Huber loss:

$$\ell_\delta(h, a) = \begin{cases} \frac{(h-a)^2}{2}, & |h - a| \leq \delta \\ \delta|h - a| - \delta^2/2, & |h - a| > \delta \end{cases} \tag{11}$$



Source: `https://www.evergreeninnovations.co/blog-machine-learning-loss-functions/`

Entropy

Entropy

Entropy $\mathbb{H}$ is a measure of the lack of predictability (uncertainty) of a random variable X with distribution p over K states:

Entropy

Entropy $\mathbb{H}$ is a measure of the lack of predictability (uncertainty) of a random variable X with distribution p over K states:

$$\mathbb{H}(X)$$

Entropy

Entropy $\mathbb{H}$ is a measure of the lack of predictability (uncertainty) of a random variable X with distribution p over K states:

$$\mathbb{H}(X) := -\sum_{k=1}^{K} p(X = k) \log_2 p(X = k)$$

Entropy

Entropy $\mathbb{H}$ is a measure of the lack of predictability (uncertainty) of a random variable X with distribution p over K states:

$$\mathbb{H}(X) := -\sum_{k=1}^{K} p(X=k)\log_2 p(X=k) = -\mathbb{E}_X[\log p(X)] \tag{12}$$

Entropy

Entropy $\mathbb{H}$ is a measure of the lack of predictability (uncertainty) of a random variable X with distribution p over K states:

$$\mathbb{H}(X) := -\sum_{k=1}^{K} p(X=k)\log_2 p(X=k) = -\mathbb{E}_X[\log p(X)] \tag{12}$$

Notes

- Entropy is measured in bits

Entropy

Entropy $\mathbb{H}$ is a measure of the lack of predictability (uncertainty) of a random variable X with distribution p over K states:

$$\mathbb{H}(X) := -\sum_{k=1}^{K} p(X = k) \log_2 p(X = k) = -\mathbb{E}_X[\log p(X)] \tag{12}$$

Notes

- Entropy is measured in bits
- For a K-ary r.v., entropy is maximized when $p(X = k) = \frac{1}{K}$

Binary entropy function

Binary entropy function

Binary r.v. $X \in \{0, 1\}$;

Binary entropy function

Binary r.v. $X \in \{0, 1\}$; $p(X = 1) = \theta$;

Binary entropy function

Binary r.v. $X \in \{0, 1\}$; $p(X = 1) = \theta$; $p(X = 0) = 1 - \theta$.

Binary entropy function

Binary r.v. $X \in \{0, 1\}$; $p(X = 1) = \theta$; $p(X = 0) = 1 - \theta$.

The binary entropy is given by:

Binary entropy function

Binary r.v. $X \in \{0, 1\}$; $p(X = 1) = \theta$; $p(X = 0) = 1 - \theta$.

The binary entropy is given by:

$$\begin{aligned} \mathbb{H}(X) &= -\sum_{k=1}^{K} p(X = k) \log_2 p(X = k) && (13) \\ &= -[p(X = 1) \log_2 p(X = 1) + p(X = 0) \log_2 p(X = 0)] && (14) \\ &= -[\theta \log_2 \theta + (1 - \theta) \log_2 (1 - \theta)] && (15) \end{aligned}$$

Multivariate entropy functions

Multivariate entropy functions

- Cross entropy between distribution p and q:

Multivariate entropy functions

- Cross entropy between distribution p and q:

$$\mathbb{H}(p, q)$$

Multivariate entropy functions

- Cross entropy between distribution p and q:

$$\mathbb{H}(p, q) := -\sum_{k=1}^{K} p_k \log q_k \tag{16}$$

Multivariate entropy functions

- Cross entropy between distribution p and q:

$$\mathbb{H}(p, q) := -\sum_{k=1}^{K} p_k \log q_k \tag{16}$$

- Joint entropy of two r.v.'s X and Y:

Multivariate entropy functions

- Cross entropy between distribution p and q:

$$\mathbb{H}(p, q) := -\sum_{k=1}^{K} p_k \log q_k \tag{16}$$

- Joint entropy of two r.v.'s X and Y:

$$\mathbb{H}(X, Y)$$

Multivariate entropy functions

- Cross entropy between distribution p and q:

$$\mathbb{H}(p, q) := -\sum_{k=1}^{K} p_k \log q_k \tag{16}$$

- Joint entropy of two r.v.'s X and Y:

$$\mathbb{H}(X, Y) := -\sum_{x,y} p(x, y) \log_2 p(x, y) \tag{17}$$

- Conditional entropy:

Multivariate entropy functions

- Cross entropy between distribution p and q:

$$\mathbb{H}(p, q) := -\sum_{k=1}^{K} p_k \log q_k \tag{16}$$

- Joint entropy of two r.v.'s X and Y:

$$\mathbb{H}(X, Y) := -\sum_{x,y} p(x, y) \log_2 p(x, y) \tag{17}$$

- Conditional entropy:

$$\mathbb{H}(Y|X)$$

Multivariate entropy functions

- Cross entropy between distribution p and q:

$$\mathbb{H}(p, q) := -\sum_{k=1}^{K} p_k \log q_k \tag{16}$$

- Joint entropy of two r.v.'s X and Y:

$$\mathbb{H}(X, Y) := -\sum_{x,y} p(x, y) \log_2 p(x, y) \tag{17}$$

- Conditional entropy:

$$\mathbb{H}(Y|X) := \mathbb{H}(X, Y) - \mathbb{H}(X) \tag{18}$$

Multivariate entropy functions

- Cross entropy between distribution p and q:

$$\mathbb{H}(p, q) := -\sum_{k=1}^{K} p_k \log q_k \tag{16}$$

- Joint entropy of two r.v.'s X and Y:

$$\mathbb{H}(X, Y) := -\sum_{x,y} p(x, y) \log_2 p(x, y) \tag{17}$$

- Conditional entropy:

$$\mathbb{H}(Y|X) := \mathbb{H}(X, Y) - \mathbb{H}(X) \tag{18}$$

- Chain rule for entropy:

Multivariate entropy functions

- Cross entropy between distribution p and q:

$$\mathbb{H}(p, q) := -\sum_{k=1}^{K} p_k \log q_k \tag{16}$$

- Joint entropy of two r.v.'s X and Y:

$$\mathbb{H}(X, Y) := -\sum_{x,y} p(x, y) \log_2 p(x, y) \tag{17}$$

- Conditional entropy:

$$\mathbb{H}(Y|X) := \mathbb{H}(X, Y) - \mathbb{H}(X) \tag{18}$$

- Chain rule for entropy:

$$\mathbb{H}(X_1, X_2, \ldots, X_n) = \sum_{i=1}^{n} \mathbb{H}(X_i|X_1, \ldots, X_{i-1}) \tag{19}$$

Relative entropy

Relative entropy

Also known as the **Kullback-Leibler (KL) divergence** or **information gain**.

Relative entropy

Also known as the **Kullback-Leibler (KL) divergence** or **information gain**.

It measures the dissimilarity (distance) between two distributions p and q:

Relative entropy

Also known as the **Kullback-Leibler (KL) divergence** or **information gain**.

It measures the dissimilarity (distance) between two distributions p and q:

$$\mathbb{KL}(p||q) := \sum_{k=1}^{K} p_k \log \frac{p_k}{q_k} \quad \text{(Discrete)} \tag{20}$$

$$\mathbb{KL}(p||q) := \int p_k \log \frac{p_k}{q_k} dx \quad \text{(Continuous)} \tag{21}$$

In discrete case, we can show that:

$$\mathbb{KL}(p||q) = \mathbb{H}(p,q) - \mathbb{H}(p) \tag{22}$$

i.e. cross entropy (between p and q minus entropy of p).

Relative entropy

Also known as the **Kullback-Leibler (KL) divergence** or **information gain**.

It measures the dissimilarity (distance) between two distributions p and q:

$$\mathbb{KL}(p||q) := \sum_{k=1}^{K} p_k \log \frac{p_k}{q_k} \quad \text{(Discrete)} \tag{20}$$

$$\mathbb{KL}(p||q) := \int p_k \log \frac{p_k}{q_k} dx \quad \text{(Continuous)} \tag{21}$$

Relative entropy

Also known as the **Kullback-Leibler (KL) divergence** or **information gain**.

It measures the dissimilarity (distance) between two distributions p and q:

$$\mathbb{KL}(p||q) := \sum_{k=1}^{K} p_k \log \frac{p_k}{q_k} \quad \text{(Discrete)} \tag{20}$$

$$\mathbb{KL}(p||q) := \int p_k \log \frac{p_k}{q_k} dx \quad \text{(Continuous)} \tag{21}$$

In discrete case, we can show that:

Relative entropy

Also known as the **Kullback-Leibler (KL) divergence** or **information gain**.

It measures the dissimilarity (distance) between two distributions p and q:

$$\mathbb{KL}(p||q) := \sum_{k=1}^{K} p_k \log \frac{p_k}{q_k} \quad \text{(Discrete)} \tag{20}$$

$$\mathbb{KL}(p||q) := \int p_k \log \frac{p_k}{q_k} dx \quad \text{(Continuous)} \tag{21}$$

In discrete case, we can show that:

$$\mathbb{KL}(p||q) = \mathbb{H}(p, q) - \mathbb{H}(p) \tag{22}$$

Relative entropy

Also known as the **Kullback-Leibler (KL) divergence** or **information gain**.

It measures the dissimilarity (distance) between two distributions p and q:

$$\mathbb{KL}(p||q) := \sum_{k=1}^{K} p_k \log \frac{p_k}{q_k} \quad \text{(Discrete)} \tag{20}$$

$$\mathbb{KL}(p||q) := \int p_k \log \frac{p_k}{q_k} dx \quad \text{(Continuous)} \tag{21}$$

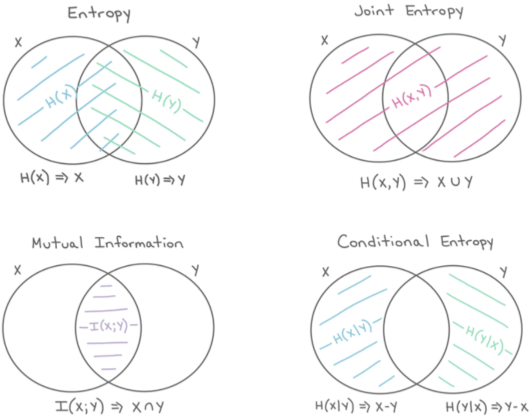
In discrete case, we can show that:

$$\mathbb{KL}(p||q) = \mathbb{H}(p, q) - \mathbb{H}(p) \tag{22}$$

i.e. cross entropy (between p and q minus entropy of p).

Entropy Venn diagrams

Entropy Venn diagrams



Source: PMLI Figure 6.4, page 211

Mutual information (MI)

Mutual information (MI)

This measures the dependency between two r.v.'s (more robust than correlation):

Mutual information (MI)

This measures the dependency between two r.v.'s (more robust than correlation):

$$\mathbb{I}(X;Y) := \mathbb{KL}(p(x,y)||p(x)p(y))$$

Mutual information (MI)

This measures the dependency between two r.v.'s (more robust than correlation):

$$\mathbb{I}(X;Y) := \mathbb{KL}(p(x,y)||p(x)p(y)) = \sum_{y \in Y} \sum_{x \in X} p(x,y) \log \frac{p(x,y)}{p(x)p(y)} \tag{23}$$

Mutual information (MI)

This measures the dependency between two r.v.'s (more robust than correlation):

$$\mathbb{I}(X;Y) := \mathbb{KL}(p(x,y)||p(x)p(y)) = \sum_{y\in Y}\sum_{x\in X} p(x,y)\log\frac{p(x,y)}{p(x)p(y)} \tag{23}$$

- Can also be written as:

Mutual information (MI)

This measures the dependency between two r.v.'s (more robust than correlation):

$$\mathbb{I}(X;Y) := \mathbb{KL}(p(x,y)||p(x)p(y)) = \sum_{y\in Y}\sum_{x\in X} p(x,y)\log\frac{p(x,y)}{p(x)p(y)} \tag{23}$$

- Can also be written as:

$$\mathbb{I}(X;Y) = \mathbb{H}(X) - \mathbb{H}(X|Y) = \mathbb{H}(Y) - \mathbb{H}(Y|X) \tag{24}$$

Mutual information (MI)

This measures the dependency between two r.v.'s (more robust than correlation):

$$\mathbb{I}(X;Y) := \mathbb{KL}(p(x,y)||p(x)p(y)) = \sum_{y\in Y}\sum_{x\in X} p(x,y)\log\frac{p(x,y)}{p(x)p(y)} \tag{23}$$

- Can also be written as:

$$\mathbb{I}(X;Y) = \mathbb{H}(X) - \mathbb{H}(X|Y) = \mathbb{H}(Y) - \mathbb{H}(Y|X) \tag{24}$$

- MI is always ≥ 0.

Mutual information (MI)

This measures the dependency between two r.v.'s (more robust than correlation):

$$\mathbb{I}(X;Y) := \mathbb{KL}(p(x,y)||p(x)p(y)) = \sum_{y \in Y} \sum_{x \in X} p(x,y) \log \frac{p(x,y)}{p(x)p(y)} \tag{23}$$

- Can also be written as:

$$\mathbb{I}(X;Y) = \mathbb{H}(X) - \mathbb{H}(X|Y) = \mathbb{H}(Y) - \mathbb{H}(Y|X) \tag{24}$$

- MI is always ≥ 0.
- A normalized estimate of MI is the "maximal information coefficient" (MIC):

Mutual information (MI)

This measures the dependency between two r.v.'s (more robust than correlation):

$$\mathbb{I}(X;Y) := \mathbb{KL}(p(x,y)||p(x)p(y)) = \sum_{y \in Y} \sum_{x \in X} p(x,y) \log \frac{p(x,y)}{p(x)p(y)} \tag{23}$$

- Can also be written as:

$$\mathbb{I}(X;Y) = \mathbb{H}(X) - \mathbb{H}(X|Y) = \mathbb{H}(Y) - \mathbb{H}(Y|X) \tag{24}$$

- MI is always ≥ 0.
- A normalized estimate of MI is the "maximal information coefficient" (MIC):

$$MIC(X,Y) = \max_{G} \frac{\mathbb{I}((X,Y)|_G)}{\log ||G||} \tag{25}$$

where G is the set of 2d grids

Reading assignments

- **PMLI** 5.1–5.4; 6.1–6.3
- **ESL** 7.1–7.7, 7.10–12