Machine Learning Notebook

Cong Bao

Contents

Chapter 1

Introduction

1.1 About this Notebook

1.2 Policy of Use

Chapter 2

Mathematics Basics

2.1 Probability

2.1.1 Basic Rules

Three Axioms of Probability Let Ω be a sample space. A probability assigns a real number $P(X)$ to each event $X \subseteq \Omega$ in such a way that

1. $P(X) \geq 0, \forall X$
2. If $X_1, X_2, \ldots$ are pairwise disjoint events ($X_1 \cap X_2 = \emptyset,\ i \neq j,\ i, j = 1, 2, \ldots$), then $P\left(\bigcup_{i=1}^{\infty} X_i\right) = \sum_{i=1}^{\infty} P(X_i)$. (This property is called countable additivity.)
3. $P(\Omega) = 1$

Joint Probability The probability both event A and B occur. $P(X, Y) = P(X \cap Y)$.

Marginalization The probability distribution of any variable in a joint distribution can be recovered by integrating (or summing) over the other variables.

1. For continuous r.v. $P(x) = \int P(x, y)\, dy$; $P(y) = \int P(x, y)\, dx$.
2. For discrete r.v. $P(x) = \sum_y P(x, y)$; $P(y) = \sum_x P(x, y)$.
3. For mixed r.v. $P(x, y) = \sum_w \int P(w, x, y, z)\, dz$, where w is discrete and z is continuous.

Conditional Probibility $P(X = x|Y = y)$ is the probability $X = x$ occurs given the knowledge $Y = y$ occurs. Conditional probability can be extracted from joint probability that

$$P(x|y = y^*) = \frac{P(x, y = y^*)}{\int P(x, y = y^*)\, dx} = \frac{P(x, y = y^*)}{P(y = y^*)}$$

Usually, the formula is written as $P(x|y) = \frac{P(x,y)}{P(y)}$.

Product Rule The formula can be rearranged as $P\left(x,y\right) = P\left(x|y\right)P\left(y\right) = P\left(y|x\right)P\left(x\right)$. In case of multiple variables

$$\begin{aligned} P\left(w,x,y,z\right) &= P\left(w,x,y|z\right)P\left(z\right) \\ &= P\left(w,x|y,z\right)P\left(y|z\right)P\left(z\right) \\ &= P\left(w|x,y,z\right)P\left(x|y,z\right)P\left(y|z\right)P\left(z\right) \end{aligned}$$

Independence If two variables x and y are independent, then r.v. x tells nothing about r.v. y (and vice-versa)

$$\begin{aligned} P\left(x|y\right) &= P\left(x\right) \\ P\left(y|x\right) &= P\left(y\right) \\ P\left(x,y\right) &= P\left(x\right)P\left(y\right) \end{aligned}$$

Baye's Rule By rearranging formula in Product Rule, we have

$$\begin{aligned} P\left(y|x\right) &= \frac{P\left(x|y\right)P\left(y\right)}{P\left(x\right)} \\ &= \frac{P\left(x|y\right)P\left(y\right)}{\int P\left(x,y\right)dy} \\ &= \frac{P\left(x|y\right)P\left(y\right)}{\int P\left(x|y\right)P\left(y\right)dy} \end{aligned}$$

Expectation Expectation tells us the excepted or average value of some function $f(x)$, taking into account the distribution of x.

$$\begin{aligned} \mathbf{E}\left[f(x)\right] &= \sum_x f(x)P\left(x\right) \\ \mathbf{E}\left[f(x)\right] &= \int f(x)P\left(x\right)dx \end{aligned}$$

Definition in two dimensions: $\mathbf{E}\left[f(x,y)\right] = \iint f(x,y)P\left(x,y\right)dx\ dy$

Function $f(\bullet)$	Expectation
x^k	k^{th} moment about zero
$(x-\mu_x)^k$	k^{th} moment about the mean

Function $f(\bullet)$	Expectation
x	mean, μ_x
$(x-\mu_x)^2$	variance
$(x-\mu_x)^3$	skew
$(x-\mu_x)^4$	kurtosis
$(x-\mu_x)(x-\mu_y)$	covariance of x and y

Besides, Expectation has the following four rules

1. Expected value of a constant is the constant $\mathbf{E}[\kappa] = \kappa$.
2. Expected value of constant times function is constant times excepted value of function $\mathbf{E}[kf(x)] = k\mathbf{E}[f(x)]$.
3. Expectation of sum of functions is sum of expectation of functions $\mathbf{E}[f(x) + g(y)] = \mathbf{E}[f(x)] + \mathbf{E}[g(x)]$.
4. Expectation of product of functions in variables x and y is product of expectations of functions if x and y are independent $\mathbf{E}[f(x)g(y)] = \mathbf{E}[f(x)]\,\mathbf{E}[g(y)]\,, x \perp\!\!\!\perp y$.

2.1.2 Common Probability Distributions

Bernoulli Bernoulli distribution describes situation where only two possible outcomes $y = 0/y = 1$ or failure/success.

1. $P(x) = \mathbf{Bern}_x[\lambda] = \lambda^x(1-\lambda)^{1-x}$
2. univariate, discrete, binary
3. $x \in \{0, 1\}$; $\lambda \in [0, 1]$
4. $\mathbf{E}[x] = \lambda$, $\mathbf{Var}[x] = \lambda(1-\lambda)$

Beta Beta distribution is the conjugate distribution to Bernoulli distribution.

1. $P(\lambda) = \mathbf{Beta}_\lambda[\alpha, \beta] = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\lambda^{\alpha-1}(1-\lambda)^{\beta-1}$
2. univariate, continuous, unbounded
3. $\lambda \in \mathbb{R}$; $\alpha \in \mathbb{R}_+$, $\beta \in \mathbb{R}_+$
4. $\mathbf{E}[\lambda] = \frac{\alpha}{\alpha+\beta}$, $\mathbf{Var}[\lambda] = \frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}$

Categorical Categorical distribution describes situation with K possible outcomes.

1. $P(x) = \mathbf{Cat}_x[\boldsymbol{\lambda}]$, $P(x = k) = \lambda_k$, $P(\boldsymbol{x} = \boldsymbol{e}_k) = \prod_{j=1}^K \lambda_j^{x_j} = \lambda_k$
2. univariate, discrete, multi-valued
3. $x \in \{1, 2, \ldots, K\}$; $\lambda_k \in [0, 1]$ where $\sum_k \lambda_k = 1$
4. $\mathbf{E}[x_i] = \lambda_i$, $\mathbf{Var}[x_i] = \lambda_i(1-\lambda_i)$, $\mathbf{Cov}[x_i, x_j] = -\lambda_i\lambda_j\ (i \neq j)$

Dirichlet Dirichlet distribution is the conjugate distribution to categorical distribution.

1. $P(\boldsymbol{\lambda}) = \mathbf{Dir}_{\boldsymbol{\lambda}}[\boldsymbol{\alpha}] = \frac{\Gamma(\sum_{k=1}^K \alpha_k)}{\prod_{k=1}^K \Gamma(\alpha_k)} \prod_{k=1}^K \lambda_k^{\alpha_k - 1}$
2. multivariate, continuous, bounded, sums to one
3. $\boldsymbol{\lambda} = [\lambda_1, \lambda_2, \ldots, \lambda_K]^\top$, $\lambda_k \in [0, 1]$, $\sum_{k=1}^K \lambda_k = 1$; $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \ldots, \alpha_K]$, $\alpha_k \in \mathbb{R}_+$

4. $\mathbf{E}\left[\lambda_i\right] = \frac{\alpha_i}{\sum_k \alpha_k}$, $\mathbf{Var}\left[\lambda_i\right] = \frac{\alpha_i(\sum_k \alpha_k - \alpha_i)}{(\sum_k \alpha_k)^2(\sum_k \alpha_k + 1)}$, $\mathbf{Cov}\left[\lambda_i, \lambda_j\right] = \frac{-\alpha_i\alpha_j}{(\sum_k \alpha_k)^2(\sum_k \alpha_k + 1)}$ $(i \neq j)$

Univariate Normal Univariate normal distribution describes single continuous variable.

1. $P\left(x\right) = \mathbf{Norm}_x[\mu, \sigma^2] = \frac{1}{\sqrt{2\pi\sigma^2}} exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$
2. univariate, continuous, unbounded
3. $x \in \mathbb{R}$; $\mu \in \mathbb{R}$, $\sigma^2 \in \mathbb{R}_+$
4. $\mathbf{E}\left[x\right] = \mu$, $\mathbf{Var}\left[x\right] = \sigma^2$

Normal Inverse Gamma Normal inverse gamma distribution is a conjugate distribution to univariate normal distribution.

1. $P\left(\mu, \sigma^2\right) = \mathbf{NormInvGam}_{\mu,\sigma^2}[\alpha, \beta, \gamma, \delta] = \frac{\sqrt{\gamma}}{\sqrt{2\pi\sigma^2}} \frac{\beta^\alpha}{\Gamma(\alpha)} \left(\frac{1}{\sigma^2}\right)^{\alpha+1} exp\left(-\frac{2\beta+\gamma(\delta-\mu)^2}{2\sigma^2}\right)$
2. bivariate, continuous, μ unbounded, σ^2 bounded below
3. $\mu \in \mathbb{R}$, $\sigma^2 \in \mathbb{R}_+$; $\alpha \in \mathbb{R}_+$, $\beta \in \mathbb{R}_+$, $\gamma \in \mathbb{R}_+$, $\delta \in \mathbb{R}$
4. $\mathbf{E}\left[\mu\right] = \delta$, $\mathbf{E}\left[\sigma^2\right] = \frac{\beta}{\alpha-1}$ $(\alpha > 1)$, $\mathbf{Var}\left[\mu\right] = \frac{\beta}{(\alpha-1)\gamma}$ $(\alpha > 1)$, $\mathbf{Var}\left[\sigma^2\right] = \frac{\beta^2}{(\alpha-1)^2(\alpha-2)}$ $(\alpha > 2)$, $\mathbf{Cov}\left[\mu, \sigma^2\right] = 0$ $(\alpha > 1)$

Multivariate Normal Multivariate normal distribution describes multiple continuous variables. It takes two parameters: a vector containing mean position $\boldsymbol{\mu}$, and a symmetric positive definite covariance matrix $\boldsymbol{\Sigma}$.

1. $P\left(\boldsymbol{x}\right) = \mathbf{Norm}_{\boldsymbol{x}}[\boldsymbol{\mu}, \boldsymbol{\Sigma}] = \frac{1}{\sqrt{det(2\pi\boldsymbol{\Sigma})}} exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^\top\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right)$
2. multivariate, continuous, unbounded
3. $\boldsymbol{x} \in \mathbb{R}^K$; $\boldsymbol{\mu} \in \mathbb{R}^K$, $\boldsymbol{\Sigma} \in \mathbb{R}^{K\times K}$ (positive semi-definite matrix)
4. $\mathbf{E}\left[\boldsymbol{x}\right] = \boldsymbol{\mu}$, $\mathbf{Var}\left[\boldsymbol{x}\right] = \boldsymbol{\Sigma}$

Normal Inverse Wishart Normal inverse wishart distribution is a conjugate distribution to multivariate normal distribution.

1. $P\left(\boldsymbol{\mu}, \boldsymbol{\Sigma}\right) = \mathbf{NormInvWis}_{\boldsymbol{\mu},\boldsymbol{\Sigma}}[\alpha, \boldsymbol{\Psi}, \gamma, \boldsymbol{\delta}]$
   $= \frac{\gamma^{D/2}|\boldsymbol{\Psi}|^{\alpha/2}|\boldsymbol{\Sigma}|^{-\frac{\alpha+D+2}{2}}}{(2\pi)^{D/2}2^{(\alpha\boldsymbol{\Sigma})/2}\Gamma_D(\alpha/2)} exp\left(-\frac{1}{2}(\mathrm{Tr}(\boldsymbol{\Psi}\boldsymbol{\Sigma}^{-1}) + \gamma(\boldsymbol{\mu}-\boldsymbol{\delta})^\top\boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}-\boldsymbol{\delta})\right)$
2. multivariate, $\boldsymbol{\mu}$ unbounded, $\boldsymbol{\Sigma}$ square, positive definite
3. $\boldsymbol{\mu} \in \mathbb{R}^K$, $\boldsymbol{\Sigma} \in \mathbb{R}^{K\times K}$; $\alpha \in \mathbb{R}_{>D-1}$, $\boldsymbol{\Psi} \in \mathbb{R}^{K\times K}$, $\gamma \in \mathbb{R}_+$, $\boldsymbol{\delta} \in \mathbb{R}^K$

2.2 Linear Algebra

2.2.1 Basic Rules

2.2.2 Square Matrices

2.3 Calculus

2.3.1 Differentiation & Integration

2.3.2 Multivariate Calculus

2.4 Informatics

2.4.1 Entropy

2.5 Optimization

2.5.1 One-dimensional Minimization

2.5.2 Gradient Descent

2.5.3 Quadratic Functions

2.5.4 General Functions

2.5.5 Optimization with Constraints

Chapter 3

Machine Learning Basics

3.1 Regularization

3.1.1 Under-fitting & Over-fitting

Under-fitting If $N > D$ (e.g. 30 data points, 2 dimensions) we have more equations than unknowns: over-determined system. Input-output relations can only hold approximately.

Over-fitting If $N < D$ (e.g. 30points, 15265 dimensions) we have more unknowns than equations: under-determined system. Input-output equations hold exactly, but we are simply memorizing data.

3.1.2 Bias & Variance

High Bias & Low Variance A rigid model's (low complexity) performance is more predictable in the test set but the model may not be good even on the training set.

Low Bias & High Variance A flexible model (high complexity) approximates the target function well in the training set but can "overtrain" and have poor performance on the test set.

3.1.3 Vector Norm

L1, ("Manhattan") norm $||\boldsymbol{w}||_1 = \sum_{d=1}^{D} |w_d|$

L2, ("Euclidean") norm $||\boldsymbol{w}||_2 = \sqrt{\sum_{d=1}^{D} w_d^2} = \sqrt{\langle \boldsymbol{w}, \boldsymbol{w} \rangle} = \sqrt{\boldsymbol{w}^\top \boldsymbol{w}}$

Lp norm, p>1 $||\boldsymbol{w}||_p = \left(\sum_{d=1}^{D} w_d^p\right)^{1/p}$

3.1.4 Penalize Complexity

In linear regression, the residual vector is $\boldsymbol{\epsilon} = \boldsymbol{y} - \boldsymbol{\Psi}\boldsymbol{w}$. The loss function is $L(\boldsymbol{w}) = \boldsymbol{\epsilon}^\top\boldsymbol{\epsilon}$. We add a complexity term $R(\boldsymbol{w}) = ||\boldsymbol{w}||_2 = \boldsymbol{w}^\top\boldsymbol{w}$ to the loss function. Hence, the original loss function becomes $L(\boldsymbol{w}) = \boldsymbol{\epsilon}^\top\boldsymbol{\epsilon} + \lambda\boldsymbol{w}^\top\boldsymbol{w}$.

Without regularization, the loss function is $L(\boldsymbol{w}) = \boldsymbol{\epsilon}^\top\boldsymbol{\epsilon}$. Let $\nabla L(\boldsymbol{w}^*) = 0$, we have $\boldsymbol{w}^* = (\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\boldsymbol{y}$.

With L2-regularization, the loss function is $L(\boldsymbol{w}) = \boldsymbol{\epsilon}^\top\boldsymbol{\epsilon} + \lambda\boldsymbol{w}^\top\boldsymbol{w}$. Let $\nabla L(\boldsymbol{w}^*) = 0$, we have $\boldsymbol{w}^* = (\mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^\top\boldsymbol{y}$. The additional $\lambda\mathbf{I}$ makes the data matrix more robust to calculate inversion.

3.2 Cross-Validation

We can select hyperparameters with (cross-)validation. Cross-validation excludes part of the training data from parameter estimation, and use them only to predict the test error.

K-fold cross validation: split data set into K folds and each time train on (K-1) folds and valid on the remaining fold until all folds have been used as validation fold. The cross-validation error is the average of K validation errors. We pick hyperparameters that minimize cross-validation error.

3.3 Bayesian Learning

3.3.1 Bayes' Rule Terminology

Bayes' Rule:

$$P\left(y|x\right) = \frac{P\left(x|y\right)P\left(y\right)}{\int P\left(x|y\right)P\left(y\right)dy}$$

Prior $P\left(y\right)$ what we know about y before seeing x. In parameters learning we choose prior that is conjugate to likelihood.

Likelihood $P\left(x|y\right)$ propensity for observing a certain value of x given a certain value of y.

Posterior $P\left(y|x\right)$ what we know about y after seeing x. Posterior must have same form as conjugate prior distribution.

Evidence $\int P\left(x|y\right)P\left(y\right)dy$ a constant to ensure that the LHS is a valid distribution. Posterior must be a distribution which implies that evidence equals to a constant κ from conjugate relation.

3.3.2 Maximum Likelihood

Fitting As the name suggests we find the parameters under which the data $\boldsymbol{x}_{1\ldots I}$ are most likely. Here, we have assumed that data was independent.

$$\begin{aligned}\hat{\boldsymbol{\theta}} &= \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\, P\left(\boldsymbol{x}_{1\ldots I}|\boldsymbol{\theta}\right)\\ &= \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \prod_{i=1}^{I} P\left(\boldsymbol{x}_i|\boldsymbol{\theta}\right)\end{aligned}$$

Predictive Density Evaluate new data point $\boldsymbol{x}^*$ under probability distribution $P\left(\boldsymbol{x}^*|\hat{\boldsymbol{\theta}}\right)$ with best parameters.

3.3.3 Maximum a Posterior (MAP)

Fitting As the name suggests we find the parameters which maximize the posterior probability $P\left(\boldsymbol{\theta}|\boldsymbol{x}_{1\ldots I}\right)$. Again we have assumed that data was independent.

$$\begin{aligned}\hat{\boldsymbol{\theta}} &= \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\, P\left(\boldsymbol{\theta}|\boldsymbol{x}_{1\ldots I}\right)\\ &= \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\, \frac{P\left(\boldsymbol{x}_{1\ldots I}|\boldsymbol{\theta}\right) P\left(\boldsymbol{\theta}\right)}{P\left(\boldsymbol{x}_{1\ldots I}\right)}\\ &= \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\, \frac{\prod_{i=1}^{I} P\left(\boldsymbol{x}_i|\boldsymbol{\theta}\right)\ P\left(\boldsymbol{\theta}\right)}{P\left(\boldsymbol{x}_{1\ldots I}\right)}\end{aligned}$$

Since the denominator does not depend on the parameters we can instead maximize

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \prod_{i=1}^{I} P\left(\boldsymbol{x}_i|\boldsymbol{\theta}\right)\ P\left(\boldsymbol{\theta}\right)$$

Predictive Density Evaluate new data point $\boldsymbol{x}^*$ under probability distribution with MAP parameters $P\left(\boldsymbol{x}^*|\hat{\boldsymbol{\theta}}\right)$

3.3.4 Bayesian Approach

Fitting Compute the posterior distribution over possible parameter values using Bayes' rule. Principle: There are many values that could have explained the data. Instead of picking one set of parameters, try to capture all of the possibilities.

$$P\left(\boldsymbol{\theta}|\boldsymbol{x}_{1\ldots I}\right) = \frac{\prod_{i=1}^{I} P\left(\boldsymbol{x}_i|\boldsymbol{\theta}\right)\ P\left(\boldsymbol{\theta}\right)}{P\left(\boldsymbol{x}_{1\ldots I}\right)}$$

Predictive Density (a) Each possible parameter value makes a prediction. (b) Some parameters more probable than others.

$$P\left(\boldsymbol{x}^*|\boldsymbol{x}_{1\ldots I}\right) = \int P\left(\boldsymbol{x}^*|\boldsymbol{\theta}\right) P\left(\boldsymbol{\theta}|\boldsymbol{x}_{1\ldots I}\right) d\boldsymbol{\theta}$$

Make a prediction that is an infinite weighted sum (integral) of the predictions for each parameter value ($P\left(\boldsymbol{x}^*|\boldsymbol{\theta}\right)$), where weights are the probabilities ($P\left(\boldsymbol{\theta}|\boldsymbol{x}_{1\ldots I}\right)$).

3.3.5 Example: Univariate Normal Distribution

Maximum Likelihood

Likelihood given by normal distribution pdf:

$$P\left(x|\mu,\sigma^2\right) = \mathbf{Norm}_x[\mu,\sigma^2] = \frac{1}{\sqrt{2\pi\sigma^2}} exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

Apply maximum likelihood:

$$\begin{aligned}
\hat{\mu},\hat{\sigma}^2 &= \underset{\mu,\sigma^2}{\operatorname{argmax}} P\left(x_{1\ldots I}|\mu,\sigma^2\right) \\
&= \underset{\mu,\sigma^2}{\operatorname{argmax}} \prod_{i=1}^{I} P\left(x_i|\mu,\sigma^2\right) \\
&= \underset{\mu,\sigma^2}{\operatorname{argmax}} \prod_{i=1}^{I} \mathbf{Norm}_{x_i}[\mu,\sigma^2] \\
&= \underset{\mu,\sigma^2}{\operatorname{argmax}} \sum_{i=1}^{I} \log \mathbf{Norm}_{x_i}[\mu,\sigma^2] \\
&= \underset{\mu,\sigma^2}{\operatorname{argmax}} \left(-\frac{I}{2}\log 2\pi - \frac{I}{2}\log\sigma^2 - \frac{1}{2}\sum_{i=1}^{I}\frac{(x_i-\mu)^2}{\sigma^2}\right)
\end{aligned}$$

Let $\nabla L(\hat{\mu},\hat{\sigma}^2) = 0$, we have the solution:

$$\begin{aligned}
\hat{\mu} &= \frac{\sum_{i=1}^{I} x_i}{I} \\
\hat{\sigma}^2 &= \sum_{i=1}^{I}\frac{(x_i-\hat{\mu})^2}{I}
\end{aligned}$$

Maximum a Posterior

Likelihood given by normal distribution pdf:

$$P\left(x|\mu,\sigma^2\right) = \mathbf{Norm}_x[\mu,\sigma^2] = \frac{1}{\sqrt{2\pi\sigma^2}} exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

Prior given by normal inverse gamma distribution pdf:

$$P\left(\mu,\sigma^2\right) = \mathbf{NormInvGam}_{\mu,\sigma^2}[\alpha,\beta,\gamma,\delta] = \frac{\sqrt{\gamma}}{\sqrt{2\pi\sigma^2}}\frac{\beta^\alpha}{\Gamma(\alpha)}\left(\frac{1}{\sigma^2}\right)^{\alpha+1} exp\left(-\frac{2\beta+\gamma(\delta-\mu)^2}{2\sigma^2}\right)$$

Apply maximum a posterior:

$$\begin{aligned}\hat{\mu},\hat{\sigma}^2 &= \underset{\mu,\sigma^2}{\operatorname{argmax}}\prod_{i=1}^{I} P\left(x_i|\mu,\sigma^2\right)\ P\left(\mu,\sigma^2\right)\\ &= \underset{\mu,\sigma^2}{\operatorname{argmax}}\prod_{i=1}^{I}\mathbf{Norm}_{x_i}[\mu,\sigma^2]\ \mathbf{NormInvGam}_{\mu,\sigma^2}[\alpha,\beta,\gamma,\delta]\\ &= \underset{\mu,\sigma^2}{\operatorname{argmax}}\left(\sum_{i=1}^{I}\log\mathbf{Norm}_{x_i}[\mu,\sigma^2]+\log\mathbf{NormInvGam}_{\mu,\sigma^2}[\alpha,\beta,\gamma,\delta]\right)\end{aligned}$$

Let $\nabla L(\hat{\mu},\hat{\sigma}^2)=0$, we have the solution:

$$\begin{aligned}\hat{\mu} &= \frac{\sum_{i=1}^{I}x_i+\gamma\delta}{I+\gamma}\\ \hat{\sigma}^2 &= \frac{\sum_{i=1}^{I}(x_i-\mu)^2+2\beta+\gamma(\delta-\mu)^2}{I+3+2\alpha}\end{aligned}$$

Bayesian Approach

Compute the posterior distribution using Bayes' rule:

$$\begin{aligned}P\left(\mu,\sigma^2|x_{1\ldots I}\right) &= \frac{\prod_{i=1}^{I}P\left(x_i|\mu,\sigma^2\right)\ P\left(\mu,\sigma^2\right)}{P\left(x_{1\ldots I}\right)}\\ &= \frac{\prod_{i=1}^{I}\mathbf{Norm}_{x_i}[\mu,\sigma^2]\ \mathbf{NormInvGam}_{\mu,\sigma^2}[\alpha,\beta,\gamma,\delta]}{P\left(x_{1\ldots I}\right)}\\ &= \frac{\kappa(\alpha,\beta,\gamma,\delta,x_{1\ldots I})\mathbf{NormInvGam}_{\mu,\sigma^2}[\tilde{\alpha},\tilde{\beta},\tilde{\gamma},\tilde{\delta}]}{P\left(x_{1\ldots I}\right)}\\ &= \mathbf{NormInvGam}_{\mu,\sigma^2}[\tilde{\alpha},\tilde{\beta},\tilde{\gamma},\tilde{\delta}]\end{aligned}$$

where

$$\begin{aligned}\tilde{\alpha} &= \alpha+\frac{I}{2}\\ \tilde{\beta} &= \frac{\sum_i x_i^2}{2}+\beta+\frac{\gamma\delta^2}{2}-\frac{(\gamma\delta+\sum_i x_i)^2}{2(\gamma+I)}\\ \tilde{\gamma} &= \gamma+I\\ \tilde{\delta} &= \frac{\gamma\delta+\sum_i x_i}{\gamma+I}\end{aligned}$$

Take weighted sum of predictions from different parameter values:

$$
\begin{aligned}
P\left(x^*|x_{1\dots I}\right) &= \iint P\left(x^*|\mu,\sigma^2\right) P\left(\mu,\sigma^2|x_{1\dots I}\right) d\mu d\sigma \\
&= \iint \mathbf{Norm}_{x^*}[\mu,\sigma^2]\mathbf{NormInvGam}_{\mu,\sigma^2}[\tilde{\alpha},\tilde{\beta},\tilde{\gamma},\tilde{\delta}]d\mu d\sigma \\
&= \iint \kappa(\alpha,\beta,\gamma,\delta,x_{1\dots I})\mathbf{NormInvGam}_{\mu,\sigma^2}[\breve{\alpha},\breve{\beta},\breve{\gamma},\breve{\delta}]d\mu d\sigma \\
&= \kappa(\alpha,\beta,\gamma,\delta,x_{1\dots I}) \iint \mathbf{NormInvGam}_{\mu,\sigma^2}[\breve{\alpha},\breve{\beta},\breve{\gamma},\breve{\delta}]d\mu d\sigma \\
&= \kappa(\alpha,\beta,\gamma,\delta,x_{1\dots I}) \\
&= \frac{1}{\sqrt{2\pi}}\frac{\sqrt{\tilde{\gamma}}\tilde{\beta}^{\tilde{\alpha}}}{\sqrt{\breve{\gamma}}\breve{\beta}^{\breve{\alpha}}}\frac{\Gamma(\breve{\alpha})}{\Gamma(\tilde{\alpha})}
\end{aligned}
$$

where

$$
\begin{aligned}
\breve{\alpha} &= \tilde{\alpha} + \frac{1}{2} \\
\breve{\beta} &= \frac{x^{*2}}{2} + \tilde{\beta} + \frac{\tilde{\gamma}^{\tilde{\delta}^2}}{2} - \frac{(\tilde{\gamma}^{\tilde{\delta}} + x^*)^2}{2(\tilde{\gamma}+1)} \\
\breve{\gamma} &= \tilde{\gamma} + 1
\end{aligned}
$$

3.3.6 Example: Categorical Distribution

Maximum Likelihood

Likelihood given by categorical distribution pdf:

$$
P\left(x|\boldsymbol{\lambda}\right) = \mathbf{Cat}_x[\boldsymbol{\lambda}] = \prod_{j=1}^{K} \lambda_j^{x_j} = \lambda_k
$$

Apply maximum likelihood:

$$
\begin{aligned}
\hat{\boldsymbol{\lambda}} &= \underset{\boldsymbol{\lambda}}{\operatorname{argmax}} \prod_{i=1}^{I} P\left(x_i|\boldsymbol{\lambda}\right) && s.t. \sum_k \lambda_k = 1 \\
&= \underset{\boldsymbol{\lambda}}{\operatorname{argmax}} \prod_{i=1}^{I} \mathbf{Cat}_{x_i}[\boldsymbol{\lambda}] && s.t. \sum_k \lambda_k = 1 \\
&= \underset{\boldsymbol{\lambda}}{\operatorname{argmax}} \prod_{k=1}^{K} \lambda_k^{N_k} && s.t. \sum_k \lambda_k = 1 \\
&= \underset{\boldsymbol{\lambda}}{\operatorname{argmax}} \sum_{k=1}^{K} N_k \log \lambda_k && s.t. \sum_k \lambda_k = 1
\end{aligned}
$$

Here, N_k represents the number of times the data is classified in class k. As before, we will instead optimize log probability. Since there is a constraint $s.t. \sum_k \lambda_k = 1$, we use Lagrange multiplier to reconstruct the loss function.

$$L(\boldsymbol{\lambda}) = \sum_{k=1}^{K} N_k \log \lambda_k + v \left(\sum_{k=1}^{K} \lambda_k - 1 \right)$$

Let $\nabla L(\boldsymbol{\lambda}, v) = 0$, we have the solution:

$$\hat{\lambda}_k = \frac{N_k}{\sum_{m=1}^{K} N_m}$$

Maximum a Posterior

Likelihood given by categorical distribution pdf:

$$P\left(x|\boldsymbol{\lambda}\right) = \mathbf{Cat}_x[\boldsymbol{\lambda}] = \prod_{j=1}^{K} \lambda_j^{x_j} = \lambda_k$$

Prior given by Dirichlet distribution pdf:

$$P\left(\boldsymbol{\lambda}\right) = \mathbf{Dir}_{\boldsymbol{\lambda}}[\boldsymbol{\alpha}] = \frac{\Gamma(\sum_{k=1}^{K} \alpha_k)}{\prod_{k=1}^{K} \Gamma(\alpha_k)} \prod_{k=1}^{K} \lambda_k^{\alpha_k - 1}$$

Apply maximum a posterior:

$$\begin{aligned}
\hat{\boldsymbol{\lambda}} &= \underset{\boldsymbol{\lambda}}{\operatorname{argmax}} \prod_{i=1}^{I} P\left(x_i|\boldsymbol{\lambda}\right)\ P\left(\boldsymbol{\lambda}\right) && s.t. \sum_k \lambda_k = 1 \\
&= \underset{\boldsymbol{\lambda}}{\operatorname{argmax}}\, \mathbf{Cat}_{x_i}[\boldsymbol{\lambda}]\ \mathbf{Dir}_{\boldsymbol{\lambda}}[\boldsymbol{\alpha}] && s.t. \sum_k \lambda_k = 1 \\
&= \underset{\boldsymbol{\lambda}}{\operatorname{argmax}} \prod_{k=1}^{K} \lambda_k^{N_k} \prod_{k=1}^{K} \lambda_k^{\alpha_k - 1} && s.t. \sum_k \lambda_k = 1 \\
&= \underset{\boldsymbol{\lambda}}{\operatorname{argmax}} \prod_{k=1}^{K} \lambda_k^{N_k + \alpha_k - 1} && s.t. \sum_k \lambda_k = 1 \\
&= \underset{\boldsymbol{\lambda}}{\operatorname{argmax}} \sum_{k=1}^{K} (N_k + \alpha_k - 1) \log \lambda_k && s.t. \sum_k \lambda_k = 1
\end{aligned}$$

The loss function is very similar to maximum likelihood (same when the prior is uniform, i.e. $\alpha_{1\ldots k} = 1$). Take derivative with Lagrange multiplier, we have the solution:

$$\hat{\lambda}_k = \frac{N_k + \alpha_k - 1}{\sum_{m=1}^{K} (N_m + \alpha_m - 1)}$$

Bayesian Approach

Compute the posterior distribution using Bayes' rule:

$$
\begin{aligned}
P\left(\boldsymbol{\lambda}|x_{1\ldots I}\right) &= \frac{\prod_{i=1}^{I} P\left(x_i|\boldsymbol{\lambda}\right)\ P\left(\boldsymbol{\lambda}\right)}{P\left(x_{1\ldots I}\right)} \\
&= \frac{\prod_{i=1}^{I} \mathbf{Cat}_{x_i}[\boldsymbol{\lambda}]\ \mathbf{Dir}_{\boldsymbol{\lambda}}[\boldsymbol{\alpha}]}{P\left(x_{1\ldots I}\right)} \\
&= \frac{\kappa(\boldsymbol{\alpha}, x_{1\ldots I})\mathbf{Dir}_{\boldsymbol{\lambda}}[\tilde{\boldsymbol{\alpha}}]}{P\left(x_{1\ldots I}\right)} \\
&= \mathbf{Dir}_{\boldsymbol{\lambda}}[\tilde{\boldsymbol{\alpha}}]
\end{aligned}
$$

Compute predictive distribution:

$$
\begin{aligned}
P\left(x^*|x_{1\ldots I}\right) &= \int P\left(x^*|\boldsymbol{\lambda}\right) P\left(\boldsymbol{\lambda}|x_{1\ldots I}\right) d\boldsymbol{\lambda} \\
&= \int \mathbf{Cat}_{x^*}[\boldsymbol{\lambda}]\mathbf{Dir}_{\boldsymbol{\lambda}}[\tilde{\boldsymbol{\alpha}}]d\boldsymbol{\lambda} \\
&= \int \kappa(x^*, \tilde{\boldsymbol{\alpha}})\mathbf{Dir}_{\boldsymbol{\lambda}}[\breve{\boldsymbol{\alpha}}]d\boldsymbol{\lambda} \\
&= \kappa(x^*, \boldsymbol{\alpha})
\end{aligned}
$$

3.4 Machine Learning Models

3.4.1 Learning and Inference

In real world problems, we usually have two tasks:

1. Observe measured data, $\mathbf{x}$
2. Draw inferences from it about world, $\mathbf{w}$

and

1. When the world state $\mathbf{w}$ is *continuous*, we'll call this *regression.*
2. When the world state $\mathbf{w}$ is *discrete*, we'll call this *classification.*

We want take observations $\mathbf{x}$, and return probability distribution $P\left(\mathbf{w}|\mathbf{x}\right)$ over possible worlds compatible with data. To solve this, we need

1. A *model* that mathematically relates the visual data $\mathbf{x}$ to the world state $\mathbf{w}$. Model specifies family of relationships, particular relationship depends on parameter θ.
2. A *learning algorithm* fits parameters θ from paired training examples $\mathbf{x_i}$, $\mathbf{w_i}$.
3. An *inference algorithm* uses model to return $P\left(\mathbf{w}|\mathbf{x}\right)$ given new observation data $\mathbf{x}$.

3.4.2 Three Types of Model

We have three types of model:

1. Model contingency of the world on the data $P\left(w|x\right)$. (Discriminative Model)

2. Model joint occurrence of world and data $P\left(x, w\right)$. (Generative Model)

3. Model contingency of data on world $P\left(x|w\right)$. (Generative Model)

Within the three models, type 1 is called *Discriminative Model*. Type 2 and 3 are called *Generative Model*.

Model $P\left(w|x\right)$ - Discriminative

1. $P\left(w|x, \theta\right) = \mathbf{Distrib}_w[f(x, \theta)]$

2. How to model: (a) Choose an appropriate form for $P\left(w\right)$. (b) Make parameters a function of x. (c) Function takes parameters θ that define its shape.

3. Learning algorithm: Learn parameters θ from training data x, w.

4. Inference algorithm: Just evaluate $P\left(w|x\right)$

Model $P\left(x, w\right)$ - Generative

1. $P\left(z|\theta\right) = \mathbf{Distrib}_z[\theta]$

2. How to model: (a) Concatenate x and w to make $z = [x^\top, w^\top]^\top$. (b) Model the pdf of z. (c) pdf takes parameters θ that define its shape.

3. Learning algorithm: Learn parameters θ from training data x, w.

4. Inference algorithm: Compute $P\left(w|x\right)$ using Bayes' rule $P\left(w|x\right) = \frac{P(x,w)}{P(x)} = \frac{P(x,w)}{\int P(x,w)dw}$.

Model $P\left(x|w\right)$ - Generative

1. $P\left(x|w, \theta\right) = \mathbf{Distrib}_x[f(w, \theta)]$

2. How to model: (a) Choose an appropriate form for $P\left(x\right)$. (b) Make parameters a function of w. (c) Function takes parameters θ that define its shape.

3. Learning algorithm: Learn parameters θ from training data x, w.

4. Define prior $P\left(w\right)$ and then compute $P\left(w|x\right)$ using Bayes' rule $P\left(w|x\right) = \frac{P(x|w)P(w)}{\int P(x|w)P(w)dw}$.

3.4.3 Example: Regression

Consider a simple case:

1. We make a univariate continuous measurement x.

2. Use this to predict a univariate continuous state w.

Model $P(w|x)$ - Discriminative

1. $P(w|x, \boldsymbol{\theta}) = \mathbf{Norm}_w[\phi_0 + \phi_1 x, \sigma^2], \boldsymbol{\theta} = \{\phi_0, \phi_1, \sigma^2\}$

2. How to model: (a) Choose normal distribution over w. (b) Make mean μ linear function of x (variance constant). (c) Parameters are ϕ_0 (y-offset), ϕ_1 (slope), σ^2 (variance). This model is called *linear regression*.

3. Learning algorithm: Learn $\boldsymbol{\theta}$ from training data x, w. e.g. MAP:

$$\begin{aligned}
\hat{\boldsymbol{\theta}} &= \underset{\boldsymbol{\theta}}{\operatorname{argmax}} P(\boldsymbol{\theta}|w_{1\ldots I}, x_{1\ldots I}) \\
&= \underset{\boldsymbol{\theta}}{\operatorname{argmax}} P(w_{1\ldots I}|x_{1\ldots I}, \boldsymbol{\theta}) P(\boldsymbol{\theta}) \\
&= \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \prod_{i=1}^{I} P(w_i|x_i, \boldsymbol{\theta}) P(\boldsymbol{\theta})
\end{aligned}$$

4. Inference algorithm:Just evaluate $P(w|x)$ for new data x.

Model $P(x, w)$ - Generative

1. $P(x, w|\boldsymbol{\theta}) = \mathbf{Norm}_{x,w}[\boldsymbol{\mu}, \boldsymbol{\Sigma}], \boldsymbol{\theta} = \{\boldsymbol{\mu}, \boldsymbol{\Sigma}\}$

2. How to model: (a) Concatenate x and w to make $z = [x^\top, w^\top]^\top$. (b) Model the pdf of z as normal distribution. (c) pdf makes parameters $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ that define its shape.

3. Learning algorithm: Learn parameters θ from training data x, w.

4. Inference algorithm: Compute $P(w|x)$ using Bayes' rule $P(w|x) = \frac{P(x,w)}{P(x)} = \frac{P(x,w)}{\int P(x,w)dw}$.

Model $P(x|w)$ - Generative

1. $P(x|w, \boldsymbol{\theta}) = \mathbf{Norm}_x[\phi_0 + \phi_1 w, \sigma^2], \boldsymbol{\theta} = \{\phi_0, \phi_1, \sigma^2\}$

2. How to model: (a) Choose normal distribution over x. (b) Make mean μ linear function of w (variance constant). (c) Parameters are ϕ_0, ϕ_1, σ^2.

3. Learning algorithm: Learn $\boldsymbol{\theta}$ from training data x, w. e.g. MAP

4. Inference algorithm: Compute $P\left(w|x\right)$ using Bayes' rule $P\left(w|x\right) = \frac{P(x,w)}{P(x)} = \frac{P(x,w)}{\int P(x,w)dw}$.

3.4.4 Example: Classification

Consider a simple case:

1. We make a univariate continuous measurement x.

2. Use this to predict a discrete binary world $w \in \{0, 1\}$.

Model $P\left(w|x\right)$ - Discriminative

1. $P\left(w|x, \boldsymbol{\theta}\right) = \mathbf{Bern}_w[\sigma\left(\phi_0 + \phi_1 x\right)], \boldsymbol{\theta} = \phi_0, \phi_1$

2. How to model: (a) Choose Bernoulli distribution for $P\left(w\right)$. (b) Make parameters a sigmoid-activated function of x. (c) Function takes parameters ϕ_0 and ϕ_1. This model is called *logistic regression*.

3. Learning algorithm: Learning by standard methods, e.g. ML, MAP, Bayesian Approach.

4. Inference algorithm: Just evaluate $P\left(w|x\right)$.

Model $P\left(x, w\right)$ - Generative

Can't build this mode very easily:

1. Concatenate continuous vector x and discrete w to make z.

2. No obvious probability distribution to model joint probability of discrete and continuous.

Model $P\left(x|w\right)$ - Generative

1. $P\left(x|w, \boldsymbol{\theta}\right) = \mathbf{Norm}_x[\mu_w, \sigma_w^2], \boldsymbol{\theta} = \{\mu_0, \mu_1, \sigma_0^2, \sigma_1^2\}$

2. How to model: (a) Choose a Normal distribution for $P\left(x\right)$. (b) Make parameters a function of discrete binary w. (c) Function takes parameters μ_0, μ_1, σ_0^2, σ_1^2 that define its shape.

3. Learning algorithm: Learning by standard methods, e.g. ML, MAP, Bayesian Approach.

4. Define prior $P\left(w\right)$ and then compute $P\left(w|x\right)$ using Bayes' rule $P\left(w|x\right) = \frac{P(x|w)P(w)}{\int P(x|w)P(w)dw}$.

Chapter 4

Regression

4.1 Linear Regression

4.2 Non-linear Regression

4.3 Kernel Trick & Gaussian Processes

4.4 Sparse Linear Regression

4.5 Dual Linear Regression

4.6 Relevance Vector Regression

Chapter 5

Classification

5.1 Logistic Regression

5.2 Non-linear Logistic Regression

5.3 Kernel Trick & Gaussian Process Classification

5.4 Multi-class Classification

Chapter 6

Support Vector Machines

6.1 Geometric Margins

6.2 Primal & Dual Problems

6.3 Support Vectors

6.4 Slack Variables

6.5 Hinge Loss

6.6 Non-linear SVMs

6.7 Kernel Trick

Chapter 7

EM Algorithm

7.1 Expectation Maximization

7.2 Example: Mixture of Gaussians

7.3 Example: t-distributions

7.4 Example: Factor Analysis

Chapter 8

Boosting

8.1 Ensemble Methods

8.2 Bagging

8.3 Boosting

8.4 Adaboost

Chapter 9

Decision Tree & Random Forest

9.1 CART

9.2 ID3

9.3 C4.5

9.4 Random Forest

Chapter 10

Graphical Models & Markov Network

10.1 Graph Definitions

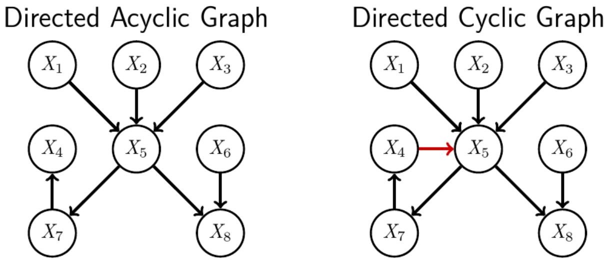
10.1.1 Graph

Graph A graph consists of nodes (vertices) and undirected or directed links (edges) between nodes.

Path A path from X_i to X_j is a sequence of connected nodes starting at X_i and ending at X_j.

10.1.2 Directed Graph

Directed Graphs Graphs that all the edges are directed.



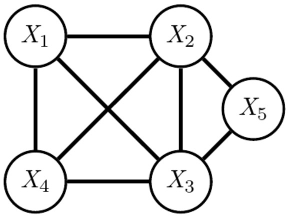
Directed Acyclic Graph Graph in which by following the direction of the arrows a node will never be visited more than once.

Parents and Children X_i is a parent of X_j if there is a link from X_i to X_j. X_i is a child of X_j if there is a link from X_j to X_i.

Ancestors and Descendants The ancestors of a node X_i are the nodes with a directed path ending at X_i. The descendants of X_i are the nodes with a directed path beginning at X_i.

10.1.3 Undirected Graph

Undirected Graph Graph that all the edges are undirected.



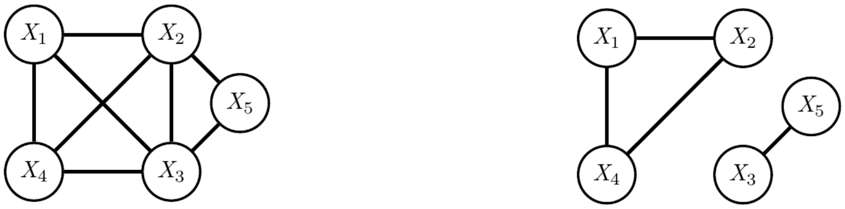
Clique A clique is a fully connected subset of nodes. (X_1, X_2, X_4) forms a (non-maximal) clique.

Maximal Clique Clique which is not a subset of a larger clique. (X_1, X_2, X_3, X_4) and (X_2, X_3, X_5) are both maximal cliques.

10.1.4 Connectivity

Connected Graph There is a path between every pair of vertices.

Connected Components In a non-connected graph, the connected components are the connected-subgraphs. (X_1, X_2, X_4) and (X_3, X_5) are the two connected components.

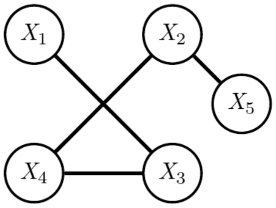


Connected Graph Connected Components

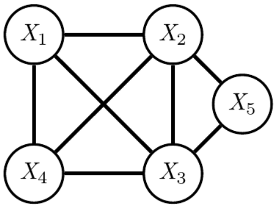
10.1.5 Connectedness

Singly-connected There is only one path from any node a to another node b.

Multiply-connected A graph is multiply-connected if it is not singly-connected.



Singly-connected



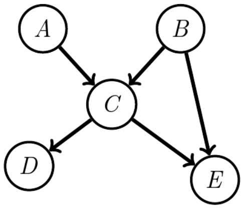
Multiply-connected

10.2 Belief Networks

10.2.1 Definition

A belief network is a directed acyclic graph in which each node is associated with the conditional probability of the node given its parents. The joint distribution is obtained by taking the product of the conditional probabilities.

$$P\left(A,B,C,D,E\right)=P\left(A\right)P\left(B\right)P\left(C|A,B\right)P\left(D|C\right)P\left(E|B,C\right)$$



$p(E|B,C)$

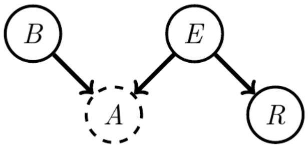
10.2.2 Uncertain Evidence

Definition In soft/uncertain evidence the variable is in more than one state, with the strength of out belief about each state being given by probabilities. For example, if y has the states $\text{dom}(y)=\{red,blue,green\}$, the vector (0.6, 0.1, 0.3) could represent the probabilities of the respective states.

Hard Evidence We are certain that a variable is in a particular state. In this state, all the probability mass is in one of the vector components, (0, 0, 1).

Inference Inference with soft-evidence can be achieved using Bayes' rule. Writing the soft-evidence as $\tilde{y}$, we have $P\left(x|\tilde{y}\right)=\sum_y P\left(x|y\right)P\left(y|\tilde{y}\right)$, where $P\left(y=i|\tilde{y}\right)$ represents the probability that y is in state i under the soft-evidence.

Jeffrey's Rule For variables x, y and $P_1(x,y)$, how do we form a joint distribution given soft-evidence $\tilde{y}$? (a) From the conditional we first define $P_1(x|y) = \frac{P_1(x,y)}{\sum_x P_1(x,y)}$. (b) Define the joint. The soft-evidence $P\left(y|\tilde{y}\right)$ then defines a new joint distribution $P_2(x,y|\tilde{y}) = P_1(x|y)P_1(y|\tilde{y})$. One can therefore view soft-evidence as defining a new joint distribution. We use a dashed circle to represent a variable in an uncertain state.



10.2.3 Independence

10.3 Markov Networks

10.4 Markov Chains

10.5 Hidden Markov Models

Appendix A

Statistical Assessment

A.1 Hypothesis Testing

A.2 Confidence Intervals