Name: Justin Park **Student ID**: 931051050

Unit Gaussian: X ~ N(μ, σ^2) Gaussian Distribution, Unit Gaussian = N(0, 1) **Uniform Distr**: prob distr that has constant prob; X ~ uniform[a, b]; f(x) = $\frac{1}{b-a}$ [ex: toss a coin] **Joint Prob**: likelihood 2 events occurring together; p(x,y) = P(X=x,Y=y); independent: p(x,y) = P(X=x)*P(Y=y) **Marg Prob**: prob of event irrespective of the outcome of another var; p(x) = $\sum_{y\in Y} p(x,y)$; ex. p(X=head) = p(X=head,Y=head) + p(X=head,Y= tail) = .25 + .25 = .5 **Bayes Rule**: $p(z|x) = \frac{p(x|z)p(z)}{p(x)}$, JP: p(x,z) = p(x|z)p(z) = p(z|x)p(x); p(z): prior, p(x): marginal, p(z|x) posterior, p(x|z): conditional **Independence**: When random variables X & Z are independent; P(x,z) = P(x) * P(z), P(x|z) = P(x), P(z|x) = P(z) **Chain in Prob**: P(x_n, x_{n-1}, … , x_2, x_1) = P(x_n|x_{n-1}, … , x_2, x_1)*P(x_{n-1}|$x_{n-2, \ldots}$, x_2, x_1) … P(x_1); Ex. P(x_2, x_1) = P(x_2|x_1)P(x_1); P(x_3, x_2, x_1) = P(x_3|x_2, x_1)P(x_2|x_1)P(x_1); When independent: P(x_n, x_{n-1}, … , x_2, x_1) = $\prod_{i=1}^{n} P$(x_i) **Exp Val**: the weighted average of possible values of a random variable; E(X) = $\sum X_i$P(X_i); X = 10, 50, 100; P = .3, .5, .2; E(X) = 10*.3 + 50*.5 + 100*.2 = 48 **Point Est** = Mean **Bias**: exp – actual = E(X) – X **Mean Squared Error (MSE) =** E(estimate – X)^2 **ED**: d(P, Q) = $\sqrt{(x_1-y_1)^2+(x_2-y_2)^2}$ **ManD**: right angle distance d(P, Q) = |x_1-y_1| + |x_2-y_2| **MinD**: d(P, Q) = $(\sum_{i=1}^{n} |x_1 - y_1|^r)^{\frac{1}{r}}$ where r=1: MD, r=2: ED, r=inf: max diff **HD**: compares two binary strings using xor, value is # of 1's 111xor100 = 2 **JD**: J(A, B) = $\frac{|A\cap B|}{|A\cup B|}$ **CS**: $\cos\theta = \frac{a\circ b}{||a||*||b||}, a^{\circ}b = a_1*b_1+a_2+b_2..etc, ||a|| = \sqrt{a_1+a_2..etc}$; Ex. A:I love data mining, B: like; I[10000],love[01000],like [00100],data [00010],mining[00001];A[11011],B[10111]; $\cos\theta = \frac{a\circ b}{||a||*||b||} = \frac{3}{4}$ **Lim**: ED: largest-scaled feature would dominate the others, HD: only binary measurement, MD: overestimate the distance, JD: does not work for ordered sequence, CS: does not work for semantical meaning **App**: ED: k-means clustering, GPS navigation, MD: A* algorithm, pac-man, HD: error detection/correction when data is transmitted over computer networks, CS: word similarity, document similarity, JD: similarity between binary vectors or sets, Z-score: outlier detection **KL Div**: measures the difference between two probability distributions; D_{KL} (q(x)||p(x)) = $\int q(x)\, log \frac{q(x)}{p(x)}$; D_{KL}(q||p) = $.5log\frac{.5}{.5}+.2log\frac{.2}{.2}+.2log\frac{.2}{.1}+.1log\frac{.1}{.2}$ **Entropy**: measure a state of disorder in the universe; $H(X) = -\sum_{i=1}^{k} P(x_i)log*P(x_i)$; Monday: [.6, .2, .2] H(X) = .6log.6 + .2log.2 + .2log.2 **Rules of Exp Val**: E(aX + b) = aE(X) + b, E(X_1 + X_2) = E(X_1) + E(X_2) **Variance**: Exp of squared diff; σ^2 = var(X) = E[(X-μ)^2]; σ^2 = var(X) = E[(X-μ)^2] = E[X^2] – (E[X])^2 → (each value – mean)^2 **Normal**: makes on a similar scale; X_{new} = $\frac{X-X_{min}}{X_{max}-X_{min}}$ **Z-score**: $X_{new} = \frac{X-mean}{std} = \frac{X-\mu}{\sigma}$ **Co-var** for two variables X_1 and X_2 = E[X_1X_2] – (E[X_1] * E[X_2]); σ_{12} > 0: + cor, σ_{12} < 0: - cor **Pear Cor**: cor between two var is the covar divided by the σ of both: $\rho_{12} = \frac{\sigma_{12}}{\sigma_1\sigma_2}, if\ \rho_{12}$>0: + cor; =0: independent; <0: – cor; **K-Means** randomly initialize the number of clusters K; Calc the ED between i and the initial points; Assign point i to the nearest cluster with minimum distance; calc the mean value of each cluster as new central points; Recluster the points based on the new means; Repeat step 3 until there are no cluster changes **Lim**: sensitive to K initial points, computation is very expensive, if there are outliers it does not work well **App to HC**: phylogenetic tree of anim evo **Lim to MIN**: sensitive to noise and outliers, **lim to HC**: once decision combine two clusters cannot be undone **Eps**: max radius of the hood **Adv to dbscan**: separate clusters of high density vs clusters of low density, handles outliers within the dataset **Super learning**: the training data is manually labelled indicating the class of the obser, new data is classified based on the training set **Unsuper learning**: the class labels of training data is unknown, given a set of measur, obser, etc. with the aim of establishing the existence of classes or clusters in the data **KNN Steps** compute distance between testing point i & labeled point; choose "K" nearest neighboring points, such as 9; use majority voting to get the label, e.g. Blue Color **Lim**: if K is too small, sensitive to noise points; if K is too large, neighborhood may include points from other classes **Adv**: local classifiers, they can produce decision boundaries of arbitrary shapes