Chapitre 5 : KNN pour l'analyse de survie :

Introduction.

Les plus proche voisins (nearest neighbour) est la technique la plus ancienne et très simple utilisée pour la classification. Selon cette technique, la classification d'un tuple de données inconnu est réalisée en analysant les classes de ses plus proches voisins. Dans le cas de l'algorithme KNN (K Nearest Neighbour), un nombre entier positif fixe k de voisins les plus proches est autorisé à voter dans le processus de classification d'un tuple de données inconnu. Lorsque k = 1, le tuple de données inconnu est classé comme la classe du tuple de données d'apprentissage qui lui est le plus proche. KNN est un apprenant paresseux non paramétrique. Contrairement aux méthodes paramétriques, la forme du modèle de classification n'est pas présumée dans le cas des méthodes non paramétriques. KNN est appelé algorithme d'apprentissage paresseux tandis que le reste des algorithmes de classification est appelé avide car il ne construit pas de classificateur juste après avoir obtenu des tuples de données d'entraînement comme cela se fait normalement dans la première étape de la classification. Précisément, il n'y a pas de phase d'entrainement explicite dans KNN. Il ne commence à fonctionner que lorsqu'il obtient un nouveau tuple pour la classification. En raison de ce principe de fonctionnement de KNN, est aussi nommé un apprenant basé sur l'instance.

Un aperçu sur l'algorithme KNN :

L'algorithme KNN classe un échantillon de test non étiqueté en fonction de la majorité des échantillons similaires parmi les k voisins les plus proches qui sont les plus proches de l'échantillon de test. Les distances entre l'échantillon de test et chacun des échantillons de données d'apprentissage sont déterminées par une mesure de distance spécifique.

Les étapes de base du classificateur KNN peuvent être décrites comme suit :

Algorithme KNN de base :

Entrée : *échantillons d'apprentissage D, échantillon de test d, K*

Sortie *: étiquette de classe de l'échantillon de test*

1 : Calculer la distance entre d et chaque échantillon de D

2 : Choisir les K échantillons de D les plus proches de d ; notons l'ensemble par P ($\in D$)

3 : Attribuer à d la classe qui est la plus fréquente (ou la classe majoritaire)

Rappel sur les mesures de distance :

La fonction de distance entre deux vecteurs x et y est une fonction d(x,y) qui définit la distance entre les deux vecteurs comme un nombre réel non négatif. Cette fonction est considérée comme une métrique si elle satisfait un certain nombre de propriétés parmi lesquelles :

1. **Non-négativité** : La distance entre x et y est toujours une valeur supérieure ou égale à zéro.
$$\forall x, y \qquad d(x, y) \geq 0$$

2. **Séparation** : La distance entre x et y est égale à zéro si et seulement si x est égal à y
$$\forall x, y \qquad d(x, y) = 0 \Leftrightarrow x = y$$

3. **Symétrie** : La distance entre x et y est égale à la distance entre y et x
$$\forall x, y \qquad d(x, y) = d(y, x)$$

4. **Inégalité triangulaire** : Compte tenu de la présence d'un troisième point z, la distance entre x et y est toujours inférieure ou égale à la somme de la distance entre x et z et de la distance entre y et z

$$\forall x, y, z \qquad d(x,y) \leq d(x,z) + d(z,y)$$

Lorsque la distance est dans l'intervalle [0,1], le calcul d'une mesure de similarité correspondante s(x,y) est le suivant : $s(x,y) = 1 - d'x, y)$.

Nous considérons les huit grandes familles de distances qui consistent en cinquante-quatre mesures de distance totales. Nous allons catégoriser ces mesures de distance en suivant une catégorisation similaire.

Mesures de distance L_p Minkowski :

Cette famille de distances comprend trois métriques de distance qui sont des cas particuliers de distance de Minkowski, correspondant à différentes valeurs de p pour cette distance de puissance. La distance de Minkowski, également connue sous le nom de norme $\boldsymbol{L_p}$, est une métrique généralisée. Il est défini comme :

$$D_{mink} = \sqrt[p]{\sum_{i=1}^{n} |x_i - y_i|^p}$$

Où p est une valeur positive. Lorsque p = 2, la distance devient la distance **euclidienne**. Lorsque p = 1, cela devient la distance de **Manhattan**. La distance de **Chebyshev** est une variante de la distance de Minkowski où p = ∞. x_i est la $i^{ème}$ valeur dans le vecteur x et y_i est la $i^{ème}$ valeur dans le vecteur y.

Manhattan (MD) :

La distance de Manhattan, également connue sous le nom de norme L_1 représente la somme des différences absolues entre les valeurs opposées dans les vecteurs.

$$MD(x,y) = \sum_{i=1}^{n} |x_i - y_i|$$

Chebyshev (CD):

La distance de Chebyshev est également appelée distance de valeur maximale, Lagrange et distance d'échiquier. Cette distance est appropriée dans les cas où deux objets doivent être définis comme différents s'ils sont différents dans une dimension quelconque [75]. C'est une métrique définie sur un espace vectoriel où la distance entre deux vecteurs est la plus grande de leur différence le long de n'importe quelle dimension de coordonnées.

$$CD(x,y) = max|xi - yi|$$

Euclidienne (ED) :

Également connue sous le nom de norme L_2 ou distance de règle, qui est une extension du théorème de Pythagore. Cette distance représente la racine de la somme du carré des différences entre les valeurs opposées dans les vecteurs.

$$ED = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

Mesures de distance L_1 :

Cette famille de distance dépend principalement de la recherche de la différence absolue, la famille comprend les distances de Lorentzian, Canberra, Sorensen, Soergel, Kulczynski, Mean Character, Non

Intersection.

Distance lorentzienne (LD) :

La distance lorentzienne est représentée par le logarithme népérien de la différence absolue entre deux vecteurs. Cette distance est sensible aux petits changements puisque l'échelle logarithmique élargit la plage inférieure et comprime la plage supérieure.

$$LD(x, y) = \sum_{i=1}^{n} \ln (1 + |x_i - y_i|)$$

Où ln est le logarithme népérien. 1 est ajouté pour s'assurer de la propriété de non-négativité et pour éviter un log de zéro.

Distance de Canberra (CanD) :

Il s'agit d'une version pondérée de la distance de Manhattan, où la différence absolue entre les valeurs d'attribut des vecteurs x et y est divisée par la somme des valeurs d'attribut absolues avant la sommation. Cette distance est principalement utilisée pour les valeurs positives. Elle est très sensible aux petits changements proches de zéro, où elle est plus sensible aux différences proportionnelles qu'aux différences absolues. Par conséquent, cette caractéristique devient plus apparente dans un espace de dimension supérieure, respectivement avec un nombre croissant de variables. La distance de Canberra est souvent utilisée pour les données dispersées autour d'une origine.

$$CanD(x, y) = \sum_{i=1}^{n} \frac{|x_i - y_i|}{|x_i| + |y_i|}$$

Distance de Sorensen (SD) :

Également connue sous le nom de Bray-Curtis, c'est l'une des mesures les plus couramment utilisées pour exprimer des relations en écologie, en sciences de l'environnement et dans des domaines connexes. Il s'agit d'une métrique de Manhattan modifiée, où les différences sommées entre les valeurs d'attributs des vecteurs x et y sont normalisées par leurs valeurs d'attributs sommées. Lorsque toutes les valeurs des vecteurs sont positives, cette mesure prend une valeur comprise entre zéro et un.

$$SD(x, y) = \frac{\sum_{i=1}^{n} |x_i - y_i|}{\sum_{i=1}^{n} |x_i + y_i|}$$

Distance de Soergel (SoD) :

La distance de Soergel est l'une des mesures de distance largement utilisées pour calculer la distance évolutive. Elle est également connue sous le nom de distance de Ruzicka. Pour les variables binaires uniquement, cette distance est identique au complément du coefficient de similarité de Tanimoto (ou Jaccard). Cette distance obéit aux quatre propriétés métriques fournies par tous les attributs ont des valeurs non négatives

$$\mathrm{SoD}(\mathrm{x},\mathrm{y}) = \frac{\sum_{\mathrm{i}=1}^{\mathrm{n}} |\mathrm{x_i} - \mathrm{y_i}|}{\sum_{\mathrm{i}=1}^{\mathrm{n}} \max(\mathrm{x_i}, \mathrm{y_i})}$$

Distance de Kulczynski (KD) :

Semblable à la distance de Soergel, mais au lieu d'utiliser le maximum, elle utilise la fonction minimum

$$SoD(x,y) = \frac{\sum_{i=1}^{n} |x_i - y_i|}{\sum_{i=1}^{n} \min(x_i, y_i)}$$

Distance moyenne des caractères (MCD) :

Également connue sous le nom de Manhattan moyenne ou distance de Gower

$$MCD(x,y) = \frac{\sum_{i=1}^{n} |x_i - y_i|}{n}$$

Distance de non intersection (NID) :

La distance de non intersection est le complément de la similarité d'intersection et est obtenue en soustrayant la similarité d'intersection d'un.

$$NID(x,y) = \frac{1}{2}\sum_{i=1}^{n} |x_i - y_i|$$

Mesures de distance du produit interne :

Les mesures de distance appartenant à cette famille sont calculées par certains produits de valeurs par paires des deux vecteurs, ce type de distances comprend : les distances Jaccard, Cosinus, Dice, Chord.

Distance de Jaccard (JacD) :

La distance de Jaccard mesure la dissemblance entre les ensembles d'échantillons, elle est un complément du coefficient de similarité de Jaccard et est obtenue en soustrayant le coefficient de Jaccard de un. Cette distance est une métrique

$$JacD(x,y) = \frac{\sum_{i=1}^{n}(x_i - y_i)^2}{\sum_{i=1}^{n} {x_i}^2 + \sum_{i=1}^{n} {y_i}^2 - \sum_{i=1}^{n} x_i y_i}$$

Distance cosinus (CosD) :

La distance cosinus, également appelée distance angulaire, est dérivée de la similarité cosinus qui mesure l'angle entre deux vecteurs, où la distance cosinus est obtenue en soustrayant la similarité cosinus de un.

$$CosD(x,y) = 1 - \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i=1}^{n} {x_i}^2}\sqrt{\sum_{i=1}^{n} {y_i}^2}}$$

Distance de dés (DicD) :

La distance de dés est dérivée de la similarité de dés, qui est un complément de la similarité de dés et est obtenue en soustrayant la similarité de dés de un. Elle peut être sensible aux valeurs proches de zéro. Cette distance n'est pas une métrique, en particulier, la propriété d'inégalité triangulaire ne tient pas. Cette distance est largement utilisée dans la recherche d'informations dans les documents et la taxonomie biologique.

$$DicD(x,y) = 1 - \frac{2\sum_{i=1}^{n} x_i y_i}{\sum_{i=1}^{n} {x_i}^2 + \sum_{i=1}^{n} {y_i}^2}$$

Distance de la corde (ChoD) :

Une modification de la distance euclidienne, qui a été introduite par Orloci pour être utilisée dans l'analyse des données sur la composition de la communauté. Il a été défini comme la longueur de la corde joignant deux points normalisés dans une hypersphère de rayon un. Cette distance est l'une des mesures de distance couramment utilisées pour regrouper des données continues.

$$ChoD(x,y) = \sqrt{2 - \frac{2\sum_{i=1}^{n} x_i y_i}{\sum_{i=1}^{n} {x_i}^2 \sum_{i=1}^{n} {y_i}^2}}$$

Mesures de distance Squared Chord :

Les distances qui appartiennent à cette famille sont obtenues en calculant la somme des géométries. La moyenne géométrique de deux valeurs est la racine carrée de leur produit. Les distances de cette famille ne peuvent pas être utilisées avec des vecteurs de caractéristiques de valeurs négatives, cette famille comprend les distances Bhattachayya, Squared Chord, Matusita, Hellinger

Distance de Bhattacharyya (BD) :

La distance de Bhattacharyya mesure la similarité de deux distributions de probabilité.

$$BD(x,y) = -\ln\left(\sum_{i=1}^{n} \sqrt{x_i y_i}\right)$$

Distance de la corde au carré (SCD) :

La distance de la corde au carré est principalement utilisée par les paléontologues et dans les études sur le pollen. Dans cette distance, la somme du carré de la différence de racine carrée à chaque point est prise le long des deux vecteurs, ce qui augmente la différence pour des caractéristiques plus dissemblables.

$$SCD(x,y) = \sum_{i=1}^{n} (\sqrt{x_i} - \sqrt{y_i})^2$$

Distance de Matusita (MatD) :

La distance de Matusita est la racine carrée de la distance de la corde au carré.

$$MatD(x,y) = \sqrt{\sum_{i=1}^{n} (\sqrt{x_i} - \sqrt{y_i})^2}$$

Distance de Hellinger (HeD) :

La distance de Hellinger aussi appelée distance de Jeffries - Matusita a été introduite en 1909 par Hellinger [38], c'est une métrique utilisée pour mesurer la similarité entre deux distributions de probabilité. Cette distance est étroitement liée à la distance de Bhattacharyya.

$$HeD(x,y) = \sqrt{2\sum_{i=1}^{n}(\sqrt{x_i} - \sqrt{y_i})^2}$$

Mesures de distance L_2 au carré :

Dans la famille de mesures de distance L_2, le carré de la différence à chaque point sur les deux vecteurs est pris en compte pour la distance totale, cette famille comprend Squared Euclidean, Clark, Neyman χ2, Pearson χ2, Squared χ2, Probabilistic Symmetric χ2, Divergence, Additive Symétrique χ2, Moyenne, Moyenne Euclidienne censurée et Distances du chi carré au carré.

Distance euclidienne au carré (SED) :

La distance euclidienne au carré est la somme des différences au carré sans prendre la racine carrée.

$$SED(x,y) = \sum_{i=1}^{n}(x_i - y_i)^2$$

Distance de Clark (ClaD) :

La distance de Clark aussi appelée coefficient de divergence a été introduite par Clark. C'est la racine carrée de la moitié de la distance de divergence.

$$ClaD(x,y) = \sqrt{\sum_{i=1}^{n}(\frac{x_i - y_i}{|x_i| + |y_i|})^2}$$

Distance de Neyman χ2 (NCSD) :

La distance de Neyman χ2 [56] est appelée une quasi-distance.

$$NCSD(x,y) = \sum_{i=1}^{n}\frac{(x_i - y_i)^2}{x_i}$$

Distance de Pearson χ2 (PCSD) :

Distance de Pearson χ2 [59], aussi appelée distance χ2.

$$PCSD(x,y) = \sum_{i=1}^{n}\frac{(x_i - y_i)^2}{y_i}$$

Distance χ2 au carré (SquD) :

Également appelée distance de discrimination triangulaire. Cette distance est une quasi-distance.

$$SquD(x,y) = \sum_{i=1}^{n}\frac{(x_i - y_i)^2}{x_i + y_i}$$

Distance probabiliste symétrique χ2 (PSCSD) :

Cette distance équivaut à la distance Sangvi χ2.

$$PSCSD(x,y) = 2\sum_{i=1}^{n}\frac{(x_i - y_i)^2}{x_i + y_i}$$

Distance de divergence (DivD) :

$$DivD(x,y) = 2\sum_{i=1}^{n}\frac{(x_i - y_i)^2}{(x_i + y_i)^2}$$

Additif Symétrique χ2 (ASCSD) :

Également connu sous le nom de divergence symétrique χ2.

$$ASCSD(x,y) = 2\sum_{i=1}^{n}\frac{(x_i - y_i)^2(x_i + y_i)}{x_i y_i}$$

Distance moyenne (AD) :

La distance moyenne, aussi appelée distance euclidienne moyenne est une version modifiée de la distance euclidienne. Là où la distance euclidienne a l'inconvénient suivant, "si deux vecteurs de données n'ont pas de valeurs d'attributs en commun, ils peuvent avoir une distance plus petite que l'autre paire de vecteurs de données contenant les mêmes valeurs d'attributs". cette distance était adoptée sous la formule,

$$AD(x,y) = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - y_i)^2}$$

Distance euclidienne censurée moyenne (MCED) :

Dans cette distance, la somme des différences au carré entre les valeurs est calculée et, pour obtenir la valeur moyenne, la valeur additionnée est divisée par le nombre total de valeurs où les valeurs des paires ne sont pas égales à zéro. Après cela, la racine carrée de la moyenne doit être calculée pour obtenir la distance finale.

$$MCEDx,y) = \sqrt{\frac{\sum_{i=1}^{n}(x_i - y_i)^2}{\sum_{i=1}^{n}\mathbb{1}_{x_i^2+y_i^2 \neq 0}}}$$

Khi carré au carré (SCSD) :

$$SCSD(x,y) = \sum_{i=1}^{n}\frac{(x_i - y_i)^2}{|x_i + y_i|}$$

Mesures de distance d'entropie de Shannon :

Les mesures de distance appartenant à cette famille sont liées à l'entropie de Shannon. Ces distances incluent Kullback-Leibler, Jeffreys, la divergence K, Topsoe, JensenShannon, les distances de différence de Jensen.

Distance de Kullback-Leibler (KLD) :

Également connue sous le nom de divergence KL, d'entropie relative ou d'écart d'information, qui mesure la différence entre deux distributions de probabilité. Cette distance n'est pas une mesure métrique, car elle n'est pas symétrique. De plus, elle ne satisfait pas la propriété d'inégalité triangulaire,

elle est donc appelé quasi-distance. La divergence de Kullback-Leibler a été utilisée dans plusieurs applications en langage naturel telles que l'expansion des requêtes, les modèles de langage et la catégorisation.

$$KLD(x,y) = \sum_{i=1}^{n} x_i ln \frac{x_i}{y_i}$$

Où n est la fonction logarithme népérienne.

Distance de Jeffreys (JefD) :

La distance de Jeffreys, aussi appelée J-divergence ou KL2-distance, est une version symétrique de la distance de Kullback-Leibler.

$$JefD(x,y) = \sum_{i=1}^{n} (x_i - y_i) ln \frac{x_i}{y_i}$$

Distance de K divergence (KDD) :

$$KDD(x,y) = \sum_{i=1}^{n} x_i ln \frac{2x_i}{x_i + y_i}$$

Distance de Topsoe (TopD) :

La distance de Topsoe, également appelée statistique d'information, est une version symétrique de la distance de Kullback-Leibler. La distance de Topsoe est le double de la divergence Jensen-Shannon. Cette distance n'est pas une métrique, mais sa racine carrée est une métrique.

$$TopD(x,y) = \sum_{i=1}^{n} x_i ln \frac{2x_i}{x_i + y_i} + \sum_{i=1}^{n} y_i ln \frac{2y_i}{x_i + y_i}$$

Distance de Jensen-Shannon (JSD) :

La distance de Jensen-Shannon est la racine carrée de la divergence de Jensen Shannon. C'est la moitié de la distance de Topsoe qui utilise la méthode moyenne pour rendre symétrique la divergence K.

$$JSD(x,y) = \frac{1}{2}[\sum_{i=1}^{n} x_i ln \frac{2x_i}{x_i + y_i} + \sum_{i=1}^{n} y_i ln \frac{2y_i}{x_i + y_i}]$$

Distance de différence de Jensen (JDD) :

La distance de différence de Jensen a été introduite par la formule,

$$JDD(x,y) = \frac{1}{2}[\sum_{i=1}^{n} \frac{x_i lnx_i + y_i lny_i}{2} - \left(\frac{x_i + y_i}{2}\right) \ln \left(\frac{x_i + y_i}{2}\right)]$$

Mesures de distance de vicissitude :

La famille de distance de vicissitude se compose de quatre distances, les distances Vicis-Wave Hedges, Vicis Symmetric, Max Symmetric $\chi 2$ et Min Symmetric $\chi 2$. Ces distances ont été générées à partir de la relation syntaxique pour les mesures de distance susmentionnées.

Distance Vicis-Wave Hedges (VWHD) :

La soi-disant "distance Wave-Hedges" a été appliquée à la récupération d'images compressées, à la récupération vidéo basée sur le contenu, à la classification des séries chronologiques, à la fidélité d'image, reconnaissance d'empreintes digitales, etc. La source de cette métrique échappe aux auteurs, malgré tous les efforts déployés autrement. Même le nom de la distance « Wave-Hedges » est remis en question.

$$VWHD(x,y) = \sum_{i=1}^{n} \frac{|x_i - y_i|}{\min(x_i, y_i)}$$

Distance symétrique de Vicis (VSD) :

La distance symétrique de Vicis est définie par trois formules, VSDF1, VSDF2, VSDF3 comme suit,

$$VSDF1(x,y) = \sum_{i=1}^{n} \frac{(x_i - y_i)^2}{\min(x_i, y_i)^2}$$

$$VSDF2(x,y) = \sum_{i=1}^{n} \frac{(x_i - y_i)^2}{\min(x_i, y_i)}$$

$$VSDF3(x,y) = \sum_{i=1}^{n} \frac{(x_i - y_i)^2}{\max(x_i, y_i)^2}$$

Distance χ2 symétrique maximale (MSCD) :

$$MSDC(x,y) = \max\left(\sum_{i=1}^{n} \frac{(x_i - y_i)^2}{x_i}, \sum_{i=1}^{n} \frac{(x_i - y_i)^2}{y_i}\right)$$

Distance minimale symétrique χ2 (MiSCSD) :

$$MSDC(x,y) = \min\left(\sum_{i=1}^{n} \frac{(x_i - y_i)^2}{x_i}, \sum_{i=1}^{n} \frac{(x_i - y_i)^2}{y_i}\right)$$

Autres mesures de distance :

Ces mesures présentent des mesures de distance utilisant plusieurs idées ou mesures à partir de mesures de distance précédentes, notamment, parmi d'autres, Moyenne (L_1, L_∞), Kumar-Johnson, Taneja, Pearson, Corrélation, Squared Pearson, Hamming, Hausdorff, statistique χ2, indice d'association de Whittaker, distances de Meehl, Motyka et Hassanat.

Distance moyenne (L_1, L_∞) (AvgD) :

La distance moyenne (L_1, L_∞) est la moyenne des distances de Manhattan et de Chebyshev.

$$AvgD(x,y) = \frac{\sum_{i=1}^{n}|x_i - y_i| + max_i|x_i - y_i|}{2}$$

Distance Kumar-Johnson (KJD):

$$KJD(x,y) = \sum_{i=1}^{n}\left(\frac{(x_i + y_i)^2}{2(x_i y_i)^{\frac{3}{2}}}\right)$$

Distance de Taneja (TanD):

$$TJD(x,y) = \sum_{i=1}^{n} \left(\frac{x_i + y_i}{2}\right) ln(\frac{x_i + y_i}{2\sqrt{x_i y_i}})$$

Distance de Pearson (PeaD) :

La distance de Pearson est dérivée du coefficient de corrélation de Pearson, qui mesure la relation linéaire entre deux vecteurs. Cette distance est obtenue en soustrayant le coefficient de corrélation de Pearson de un.

$$PeaD(x,y) = 1 - \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \times \sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

Avec, $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$.

Distance de corrélation (CorD) :

La distance de corrélation est une version de la distance de Pearson, où la distance de Pearson est mise à l'échelle afin d'obtenir une mesure de distance comprise entre zéro et un.

$$CorD(x,y) = \frac{1}{2}(1 - \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \times \sum_{i=1}^{n}(y_i - \bar{y})^2}})$$

Distance de Pearson au carré (SPeaD):

$$SPeaD(x,y) = 1 - (\frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \times \sum_{i=1}^{n}(y_i - \bar{y})^2}})^2$$

Distance de Hamming (HamD) :

La distance de Hamming est une métrique de distance qui mesure le nombre de discordances entre deux vecteurs. Elle est principalement utilisée pour les données nominales, les analyses de chaînes et de bits, et peut également être utile pour les données numériques.

$$HamD(x,y) = \sum_{i=1}^{n} \mathbb{1}_{x_i \neq y_i}$$

Distance de Hausdorff (HauD):

$$HauD(x,y) = \max\big(h(x,y), h(y,x)\big)$$

Où $h(x,y) = max_{x_i \in x} min_{y_i \in y} \big||x_i - y_i|\big|$, et || || est la norme du vecteur (par exemple norme L_2). La fonction $h(x,y)$ est appelée la distance de Hausdorff dirigée de x à y. La distance de Hausdorff HauD(x,y) mesure le degré d'inadéquation entre les ensembles x et y en mesurant l'éloignement entre chaque point x_i et y_i et vice versa.

Distance statistique χ2 (CSSD) :

La distance statistique χ2 a été utilisée pour la récupération d'images, l'histogramme, etc.

$$CSSD(x,y) = \sum_{i=1}^{n} \frac{x_i - m_i}{m_i}$$

Où $m_i = \frac{x_i + y_i}{2}$.

Distance d'association de l'indice de Whittaker (WIAD) :

L'indice de distance d'association de Whittaker a été conçu pour les données d'abondance des espèces.

$$WIAD(x,y) = \frac{1}{2}\sum_{i=1}^{n} \left| \frac{x_i}{\sum_{i=1}^{n} x_i} - \frac{y_i}{\sum_{i=1}^{n} y_i} \right|$$

Distance de Meehl (MeeD):

La distance de Meehl dépend d'un point consécutif dans chaque vecteur.

$$MeeD(x,y) = \sum_{i=1}^{n} (x_i - y_i - x_{i+1} + y_{i+1})^2$$

Distance de Motyka (MotD) :

$$MotD(x,y) = \frac{\sum_{i1}^{n} \max(x_i, y_i)}{\sum_{i=1}^{n} (x_i + y_i)}$$

Distance de Hassanat (HasD) :

C'est une distance non convexe introduite sous la formule,

$$HasD(x,y) = \sum_{i=1}^{n} D(x_i, y_i)$$

Où,

$$D(x_i,y_i) = \begin{cases} 1 - \dfrac{1 + \min(x_i,y_i)}{1 + \max(x_i,y_i)} & min(x_i,y_i) \geq 0 \\ 1 - \dfrac{1 + min(x_i,y_i) + |\min(x_i,y_i)|}{1 + \max(x_i,y_i) + |\min(x_i,y_i)|} & min(x_i,y_i) < 0 \end{cases}$$

Comme on peut le voir, la distance de Hassanat est bornée par [0,1[. Elle atteint 1 lorsque la valeur maximale tend vers l'infini en supposant que le minimum est fini, ou lorsque la valeur minimale tend vers moins l'infini en supposant que le maximum est fini.

$$\lim_{\max(A_i,B_i)\to+\infty} D(A_i,B_i) = \lim_{min(A_i,B_i)\to-\infty} D(A_i,B_i) = 1$$

En satisfaisant toutes les propriétés métriques, cette distance a été prouvée être une métrique. Dans cette métrique, quelle que soit la différence entre deux valeurs, la distance sera comprise entre 0 et 1. donc la distance maximale se rapproche de la dimension des vecteurs testés, donc l'augmentation des dimensions augmente la distance linéairement dans le pire Cas.

Variantes de KNN :

Il existe de nombreuses lacunes associées à l'algorithme KNN. En modifiant les facteurs d'influence, les performances de KNN peuvent être améliorées. Il existe de nombreuses variantes de l'algorithme KNN proposées dans différentes études qui ont tenté de surmonter ces lacunes. Nous allons décrire certaines d'entre eux.

KNN localement adaptatif :

Dans l'algorithme KNN standard, la valeur globale du paramètre d'entrée k est utilisée. Mais cet algorithme proposé suggérait d'utiliser différentes valeurs du paramètre k pour différentes parties de l'espace d'entrée. A chaque fois pour la classification d'une requête, la valeur de k est déterminée via l'application d'une validation croisée dans son voisinage local.

KNN ajusté au poids :

Dans l'algorithme KNN standard, tous les attributs ont la même importance. Ils donnent une contribution égale pour la classification des nouveaux tuples. Mais tous les attributs de l'ensemble de données n'ont pas la même importance. Un algorithme KNN ajusté par poids a été proposé, il apprend d'abord des poids pour différents attributs et selon les poids attribués, chaque attribut affecterait le processus de classification uniquement.

KNN amélioré pour la catégorisation de texte :

Comme nous savons à quel point la valeur du paramètre d'entrée k influence les performances de l'algorithme KNN. Il est donc crucial de choisir la valeur appropriée du paramètre k. En général, les classes ne sont pas uniformément réparties dans l'ensemble de données. Par conséquent, l'utilisation d'une valeur fixe de k pour toutes les classes entraînerait un biais vers la classe qui a le plus grand nombre de tuples. Ceci a été traité en utilisant différentes valeurs de k pour différentes classes en fonction de leur distribution de classe. Un plus grand nombre de tuples est utilisé pour classer un nouveau tuple dans une classe qui a des tuples plus nombreux.

KNN adaptatif :

Plutôt que d'utiliser une valeur fixe de k, KNN adaptatif utilise un nombre non fixe de plus proches voisins. Une grande valeur du paramètre k augmenterait également le coût et le temps de calcul en cas de grands ensembles de données. Pour résoudre ce problème, il applique trois heuristiques afin qu'une rupture précoce de l'algorithme soit possible. Ces heuristiques sur la réalisation d'une condition fixe sortiraient de l'algorithme.

Cela permettrait d'économiser du temps de calcul de l'algorithme.

KNN avec des voisins communs :

Cet algorithme propose une autre variante de KNN qui utilise les plus proches voisins partagés pour classer les documents. Pour trouver les voisins d'un nouveau tuple, il utilise la mesure de similarité BM25. Un seuil est défini, seul ce nombre de voisins les plus proches peut voter pour la classification d'un tuple inconnu.

KNN avec K-moyennes (K-Means) :

L'un des défauts de l'algorithme KNN est sa grande complexité de calcul. Pour pallier cet inconvénient en a combinant l'algorithme KNN avec l'algorithme de clustering K-Means. Dans l'algorithme proposé, les clusters des différentes catégories de l'ensemble de données d'apprentissage sont d'abord formés. Les centres de ces clusters nouvellement formés agiront désormais comme de nouveaux échantillons d'apprentissage. Pour classer un tuple inconnu, la distance de celui-ci est calculée avec ces nouveaux tuples d'apprentissage et il sera affecté à la classe du centre avec lequel le tuple a le moins de distance. L'avantage de cette variante de KNN est qu'il n'est pas nécessaire de passer le paramètre d'entrée k comme nous devons le faire dans le KNN standard.

KNN avec Mahalanobis métrique :

Les performances de l'algorithme KNN dépendent en grande partie de la métrique de distance utilisée pour trouver la distance entre deux tuples. Une nouvelle métrique de distance appelée métrique de distance de Mahalanobis a été introduite. Elle transforme tout l'espace d'entrée à l'aide d'une transformation linéaire. Dans cet espace d'entrée transformé, la distance euclidienne est la même que la

distance de Mahalanobis entre deux points de données. La distance euclidienne est la distance entre deux points quelconques, tandis que la distance de Mahalanobis est une distance entre un point et une distribution. Si le point représente la moyenne de la distribution, la distance de Mahalanobis serait nulle. Le principal avantage de prendre la métrique de distance de Mahalanobis au lieu de la métrique de distance euclidienne est qu'elle prend également en compte la corrélation entre les tuples de données.

KNN généralisé :

L'algorithme KNN n'est pas seulement utilisé pour la prédiction des attributs de classe catégorique, mais également des attributs de classe à valeur continue. Dans ce dernier cas, la moyenne des valeurs de l'attribut de classe est affectée à l'attribut de classe du tuple inconnu.

KNN informatif :

Cet algorithme introduit une nouvelle mesure appelée informativité. Cette mesure tient compte du fait que tous les k plus proches voisins n'ont pas la même importance. L'algorithme prendrait deux paramètres d'entrée au lieu d'un, c'est-à-dire k et I. la valeur de I décidera du nombre de tuples de données informatives à prendre en compte pour la classification d'un tuple inconnu. L'algorithme proposé trouve d'abord les k tuples de données les plus proches du tuple de test, puis il calcule l'informativité de ces k tuples de données. La classe majoritaire des I tuples de données les plus informatifs serait la classe du tuple de données inconnues.

KNN bayésien :

Le classificateur bayésien est l'un des algorithmes de classification qui donne une assez bonne précision. Plutôt que de donner la classe du tuple de test en sortie, le classificateur bayésien donne l'appartenance du tuple aux classes sous forme de probabilités. Pour augmenter les performances de l'algorithme KNN dans le classement, une combinaison de classificateur bayésien et de KNN a été proposée. Dans cet algorithme, initialement les k voisins les plus proches des tuples de test sont déterminés. Après cela, ces k tuples de données sont utilisés pour former le classifieur. Le classificateur donnerait comme résultat les probabilités d'appartenance du tuple de test dans les classes. Ces probabilités sont utilisées pour classer les instances.

SVM-KNN :

La machine à vecteurs de support (SVM) est une technique de classification qui peut être utilisée à la fois sur des données linéaires et non linéaires. Cette version hybride de KNN avec SVM est utilisée pour la reconnaissance visuelle des catégories. Dans l'algorithme SVM-KNN, les k voisins les plus proches des tuples inconnus sont utilisés pour former SVM. Pour la mise en œuvre de cet algorithme hybride, les k tuples de données les plus proches sont d'abord déterminés. Après, les distances par paires entre ces k tuples de données sont calculées. La matrice de noyau est calculée à partir de cette matrice de distances obtenue. Cette matrice de noyau calculée est donnée en entrée au classificateur SVM. La sortie serait la classe du tuple inconnu.

Quelques autres extensions pour KNN :

D’autres extensions pour l'algorithme KNN ont été proposées. Il s’agit de classificateur KNN basé sur la densité, le classificateur KNN variable k, le classificateur KNN pondéré, le classificateur KNN basé sur les classes, le classificateur KNN de discernabilité. Au lieu de simplement compter le nombre de voisins, le classificateur KNN basé sur la densité prend en compte un autre facteur, à savoir la densité. Le classificateur variable k KNN sélectionne différentes valeurs de k pour différents ensembles de données d'apprentissage. Le classificateur KNN pondéré calcule les poids pour toutes les caractéristiques de l'ensemble de données. Le classificateur KNN basé sur les classes sélectionne différents k pour des classes distinctes en fonction du nombre de tuples qu'il contient. Le classificateur KNN de discernabilité utilise le concept de discernabilité qui mesure la façon dont les classes distinctes d'un ensemble de données sont facilement distinguées.

Une méthode de prédiction de la probabilité de survie des K plus proches voisins

Nous allons maintenant introduire une méthode non paramétrique simple pour prédire la probabilité de survie dans le cadre de données censurées à droite sans risques concurrents. Cette méthode construit des courbes de survie de Kaplan–Meier sur la base de la survie observée des K plus proches voisins dans un ensemble d'apprentissage. La distance entre les points de données est mesurée avec une métrique sur les covariables associées aux observations.

Le choix initial pour la métrique est la distance de Mahalanobis, une métrique largement applicable avec l'avantage de n'imposer aucune hypothèse structurelle sur les données.

Description de la méthode :

Étant donné une nouvelle observation j, cette méthode génère une prédiction pour la courbe de survie de j en créant une courbe de Kaplan-Meier à partir des temps de survie (éventuellement censurés) d'observations "similaires" à partir d'un ensemble de données existant S.

Chaque observation $i \in S$ est associée à un ensemble de covariables $x_i \in \mathbb{R}^p$, un temps d'événement $t_i > 0$ et un indicateur de censure δ_i, où 0/1 indique une observation censurée/non censurée. L'ensemble de covariables x_j est associé à la nouvelle observation j. Soit $d(x_i, x_j)$ une métrique sur x qui mesure à quel point les observations i et j sont similaires. Les K plus proches voisins de j dans S sont choisis selon $d(x_i, x_j)$ pour former l'ensemble $S_j^K \subseteq S$. Un analogue pondéré de la courbe de survie de Kaplan–Meier générée à partir des observations dans S_j^K,

$$\hat{S}^K(t|x_i; w) = \prod_{i \in S_j^K : t_i < t} (1 - \frac{d_i^w}{n_i^w})$$

sert de prédiction pour la courbe de survie de l'observation j, où :

$$n_i^w = \sum_{k \in S_j^K} w\left(d(x_j, x_k)\right).I(t_k \geq t_i)$$

est le nombre pondéré d'observations dans S_j^K à risque juste avant t_i et

$$d_i^w = \sum_{k \in S_j^K} w\left(d(x_j, x_k)\right).\delta_k.I(t_k = t_i)$$

est le nombre pondéré de décès au temps t_i.

La fonction de pondération $w(.)$ pour les observations en S_j^K est non croissante dans la distance $d(.,.)$ de l'observation à x_j , mettant ainsi davantage l'accent sur des observations d'entraînement qui ressemblent davantage à la nouvelle observation. Puisque $\hat{S}^K(t|x_i; w)$ est invariante aux échelles de w, sans perte de généralité, nous exigeons que $w = 0$. Notez que le réglage $w \equiv 1$ donne une pondération égale à chacun des K plus proches voisins et donc l'estimateur original de Kaplan–Meier est récupéré.

K peut être choisi en séparant S en ensembles d'apprentissage et de validation et en effectuant des tests de validation. Les K plus proches voisins utilisés pour générer la courbe de survie prédite sont tirés de l'ensemble d'apprentissage. L'ensemble de validation est un ensemble de données d'exclusion utilisé pour tester les performances d'une plage de valeurs K et sélectionner le K le plus performant.

Pour le choix de la métrique $d(x_i, x_j)$, il existe de nombreuses options possibles, et le choix doit être orienté par le contexte du problème. Une option polyvalente est la distance de Mahalanobis.

$$d(x_i, x_j) = \sqrt{(x_i - x_j)' \Sigma_S (x_i - x_j)}$$

Où, Σ_S est la matrice de covariance de x pour les observations appartenant à S. Cette pondération inverse dans la distance de Mahalanobis rend la métrique invariante à l'échelle, et donc indépendante des unités dans lesquelles les covariables sont exprimées. Notez que cette distance peut également être appliquée aux variables catégorielles en les transformant d'abord en un ensemble de variables indicatrices.

Pour la fonction de pondération $w(.)$, plusieurs options existent, et il peut même exister un schéma de pondération optimal.

References

[1] Kartsonaki, C. (2016). Survival analysis. Diagnostic Histopathology, 22(7), 263-270..

[2] Clark, T. G., Bradburn, M. J., Love, S. B., & Altman, D. G. (2003). Survival analysis part I: basic concepts and rst analyses. British journal of cancer, 89(2), 232.

[3] Lee, Elisa T., and John Wang. Statistical methods for survival data analysis. Vol. 476. John Wiley & Sons, 2003.

[4] Allison, Paul D. Survival analysis using SAS: a practical guide. Sas Institute, 2010.

[5] Cleves, M., Gould, W., Gould, W. W., Gutierrez, R., & Marchenko, Y. (2008). An introduction to survival analysis using Stata. Stata press

[6] Jenkins, S. P. (2005). Survival analysis. Unpublished manuscript, Institute for Social and Economic Research, University of Essex, Colchester, UK, 42, 54-56

[7] Aalen, O., Borgan, O., & Gjessing, H. (2008). Survival and event history analysis: a process point of view. Springer Science & Business Media.

[8] Bewick, V., Cheek, L., & Ball, J. (2004). Statistics review 12: survival analysis. Critical care, 8(5), 389.

[9] Brostr m, G. (2012). Event history analysis with R. CRC Press.

[10] Lee, E. T., & Go, O. T. (1997). Survival analysis in public health research. Annual review of public health, 18(1), 105-134.

[11] Goel, M. K., Khanna, P., & Kishore, J. (2010). Understanding survival analysis: Kaplan-Meier estimate. International journal of Ayurveda research, 1(4), 274.

[12] Hoskin, T. (2012). Parametric and nonparametric: Demystifying the terms. In Mayo Clinic (pp. 1-5).

[13] Rich, J. T., Neely, J. G., Paniello, R. C., Voelker, C. C., Nussenbaum, B., & Wang, E. W. (2010). A practical guide to understanding Kaplan-Meier curves. Otolaryngology Head and Neck Surgery, 143(3), 331-336.

[14] Powell, J. (2008). The New Palgrave Dictionary of Economics Online.

[15] Cullen, D. J., Apolone, G., Green eld, S., Guadagnoli, E., & Cleary, P. (1994). ASA physical status and age predict morbidity after three surgical procedures. Annals of surgery, 220(1), 3.

[16] Fox, J. (2002). Cox proportional-hazards regression for survival data. An R and S-PLUS companion to applied regression, 2002.

[17] Xue, Y., & Schifano, E. D. (2017). Diagnostics for the Cox model. Communications for Statistical Applications and Methods, 24(6), 583-604.

[18] Reeves, G. K., Beral, V., Bull, D., & Quinn, M. (1999). Estimating relative survival among people registered with cancer in England and Wales. British journal of cancer, 79(1), 18.

[19] Mariotto, A. B., Noone, A. M., Howlader, N., Cho, H., Keel, G. E., Garshell, J., ... & Schwartz, L. M. (2014). Cancer survival: an overview of measures, uses, and interpretation. Journal of the National Cancer Institute Monographs, 2014(49), 145-186.

[20] Pohar, M., & Stare, J. (2006). Relative survival analysis in R. Computer methods and programs in biomedicine, 81(3), 272-278.

[21] Bajpai, Ram & Chaturvedi, Himanshu & Pandey, Arvind. (2014). Relative Survival: A Useful Tool in Population Based Health Studies. American Journal of Mathematics and Statistics 2014, 4(1): 38-45. 4. 38-45. 10.5923/j.ajms.20140401.06.

[22] Perme, M. P., & Pavlic, K. (2018). Nonparametric Relative Survival Analysis with the R Package relsurv. Journal of Statistical Software, 87(1), 1-27.

[23] Introduction à l'analyse des durées de survie Philippe SAINT PIERRE 1 Avril 2021

[24] Arbres de Décision Ricco RAKOTOMALALA Laboratoire ERIC Université Lumière Lyon 2

[25] Survey on KNN and Its Variants Alka Lamba1, Dharmender Kumar2 Student, Department of Computer Science and Engineering, GJU S&T, Hisar, India 1 Associate Professor, Department of Computer Science and Engineering, GJU S&T, Hisar, India.

[26] Optimal Survival Trees Dimitris Bertsimas · Jack Dunn · Emma Gibson Agni Orfanoudaki.

[27] David G. Kleinbaum Mitchel Klein Survival Analysis A Self-Learning Text Third Edition