Cancer Prognosis and Prediction

September 30,2019

What is Cancer?

Cancer is a group of diseases involving abnormal cell growth with the potential to invade or spread to other parts of the body.

How Cancer begins?

Cells are the basic units that make up the human body. Cells grow and divide to make new cells as the body needs them. Usually, cells die when they get too old or damaged. Then, new cells take their place.

Cancer begins when genetic changes interfere with this orderly process. Cells start to grow uncontrollably. These cells may form a mass called a tumor. A tumor can be cancerous or benign. A cancerous tumor is malignant, meaning it can grow and spread to other parts of the body. A benign tumor means the tumor can grow but will not spread.

Problem

- **Cancer** is a heterogeneous disease.
- The early diagnosis and prognosis of a cancer type have become a necessity in cancer research.
- The importance of classifying cancer patients into high or low risk groups has led many research teams, from the biomedical and the bioinformatics field, to study the application of machine learning (ML) methods.

Dataset

Dimension - 569 x 32

Target - ‘diagnosis’

Classes of ‘diagnosis’ - malignant (‘M’) or benign (‘B’)

id	diagnosis	radius_mean
texture_mean	perimeter_mean	area_mean
smoothness_mean	compactness_mean	concavity_mean
concave points_mean	symmetry_mean	fractal_dimension_mean
radius_se	texture_se	perimeter_se
area_se	smoothness_se	compactness_se
concavity_se	concave points_se	symmetry_se
fractal_dimension_se	radius_worst	texture_worst
perimeter_worst	area_worst	smoothness_worst
compactness_worst	concavity_worst	points_worst
symmetry_worst	fractal_dimension_worst	

Machine Learning Techniques

The main objective of ML techniques is to produce a model which can be used to perform classification, prediction, estimation or any other similar task. The most common task in learning process is classification.

We use different classification techniques to classify each patient's cancer as benign or malignant.

Machine Learning Techniques

A list of ML methods used are:

- ➔ Naive Bayes
- ➔ KNN
- ➔ MLP
- ➔ Logistic Regression
- ➔ Decision Tree
- ➔ Random Forest
- ➔ Rotation Forest

Bayes Theorem

➔ Given a hypothesis H and evidence E, Bayes' theorem states that the relationship between the probability of the hypothesis before getting the evidence P(H) and the probability of the hypothesis after getting the evidence P(H | E) is

$$P(H \mid E) = \frac{P(E \mid H)}{P(H)} P(E)$$

Bayesian Network

- ➔ A Bayesian network is a representation of a joint probability distribution of a set of random variables with a possible mutual causal relationship.
- ➔ The network consists of nodes representing the random variables,edges between pairs of nodes representing the causal relationship of these nodes, and a conditional probability distribution in each of the nodes.
- ➔ A bayesian network can be represented as $BN(S,\Theta_S)$.
- ➔ BN follows local markov condition that each node is independent of its non-descendents given its parents in S.
- ➔ Consists of two parts:

Contd.

$BN(S, \Theta_s)$

Qualitative

The structure S - a Directed Acyclic Graph (DAG)

- Nodes – random variables
- Arcs - direct dependence between variables

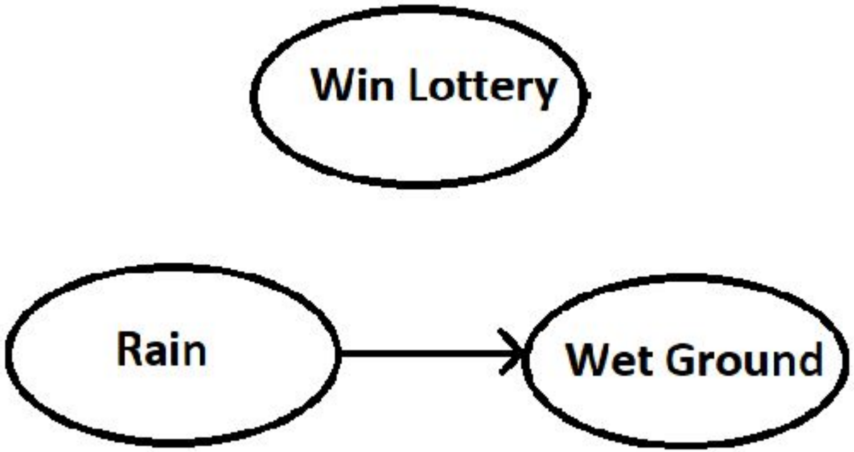
Quantitative

the set of parameters Θ_s = set of conditional probability distributions (CPDs)

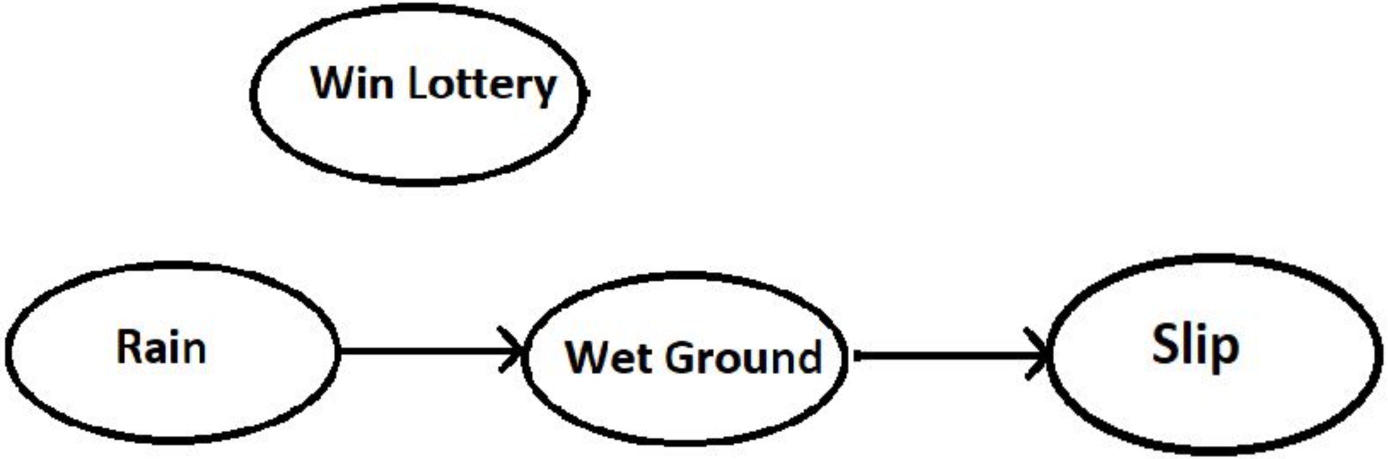
- Discrete Random Variables
- Continuous Random Variables

Example

Here,
$P(L,R,W) = P(L) \times P(R) \times P(W|R)$



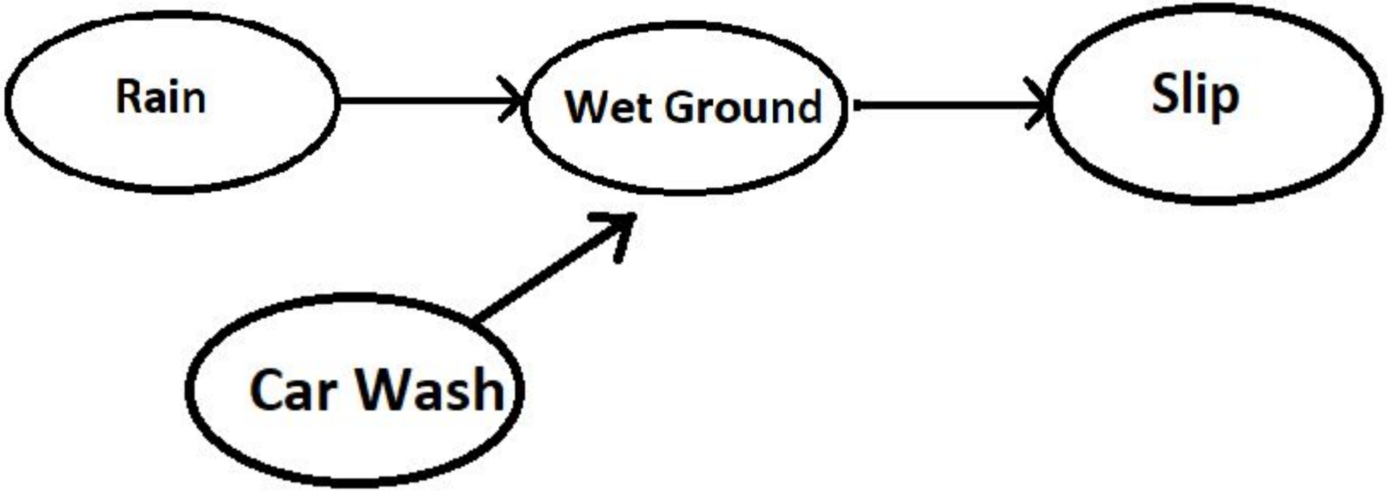
Contd.



Now, here
$P(L,R,W,S)=P(L) \times P(R) \times P(W|R) \times P(S|R,W)$

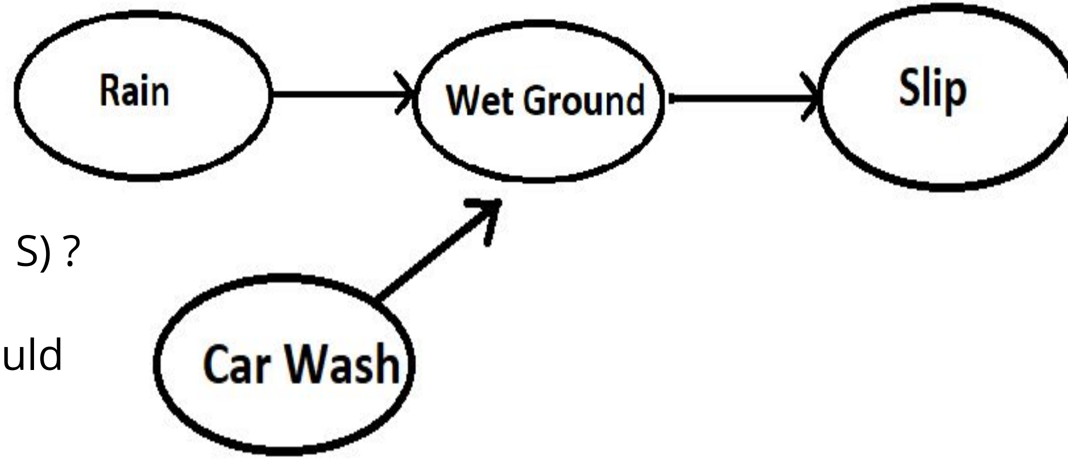
$P(L,R,W,S)=P(L) \times P(R) \times P(W|R) \times P(S|W)$

Example



P(R,W,S,C) = P(R) x P(C) x P(W| R,C) x P(W|S)

Contd.



Now What if ,we want to calculate P(R | S) ?

Through probabilistic approach ,we should

find out P(R,W,S,C) .

Now if we go through normal procedure we have to calculate 2^4 probabilities.

But through Bayesian strategy, we know the relationship exists between different variables. Therefore, through DAG

$$P(R,W,S,C) = P(R) \times P(C) \times P(W \mid R,C) \times P(W \mid S)$$

Hence the number reduces to $1+1+2^2+2^1=8<16$

Contd.

- Bayesian Networks are an efficient and effective representation of the joint probability distribution of a set of random variables.
- Algorithm takes advantage of structure to compute
  - Posterior probabilities.
  - Compute most probable instantiation.
  - Decision making.

Naive Bayes : Special Case

- It is a classification technique based on Bayes' Theorem with an assumption of independence among predictors. In simple terms, a Naive Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature.

$$P(c|x) = P(x_1|c) \times P(x_2|c) \times \ldots \times P(x_n|c) \times P(c)$$

$$c = argmax(c \mid x_1, x_2, \ldots, x_n)$$

Naive Bayes Performance

- ➔ NB is one of more simple and effective classifiers.
- ➔ NB has a very strong unrealistic independence assumption i.e., all the attributes are conditionally independent given the value of class.
- ➔ The above assumption is not true in most of the cases.
- ➔ In practice: independence assumption Bias is violated ➡ HIGH BIAS which can lead to poor classification.

Rotation Forest

- ➔ Rotation forest is a classifier ensemble based on feature extraction.
- ➔ The main heuristic consists in applying feature extraction to subsets of features and reconstructing a full feature set for each classifier in the ensemble.
- ➔ It is called Rotation Forest since PCA is a simple rotation of the coordinate axes and the base classifier model is a decision tree.

Algorithm

Let T be the number of trees required to be built.

For each tree T, perform the following steps:

- ➔ Split the attributes in the training set into K non overlapping subsets of equal size.
- ➔ We have K datasets, each with K attributes. For each of the K datasets, we proceed to do the following:
  - ◆ Bootstrap 75% of the data from each K dataset and use the bootstrapped sample for further steps.
  - ◆ Run a principal component analysis on the i^{th} subset in K. Retain all the principal components. For every feature j in the K^{th} subset, we have a principle component, a. Let's denote it as a_{ij}, where it's the principal component for the j^{th} attribute in the ith subset.

Algorithm (Contd.)

- ➔ Store the principal components for the subset.
- ➔ Create a rotation matrix of size, (n X n), where n is the total number of attributes. Arrange the principal component in the matrix such that the components match the position of the feature in the original training dataset.
- ➔ Project the training dataset on the rotation matrix using the matrix multiplication.
- ➔ Build a decision tree with the projected dataset.
- ➔ Store the tree and rotation matrix.

Results

Model	Max F1-Score	Avg F1-Score
Naive Bayes	0.9652	0.9331
KNN	0.9738	0.9357
MLP	0.9649	0.8765
Logistic Regression	0.9825	0.9484
Decision Tree	0.9738	0.9223
Random Forest	0.9826	0.9528
Rotation Forest	0.9913	0.9652

References

- ➔ Cancer classification using Rotation Forest by Kun-Hong Liua, De-Shuang Huanga
- ➔ Rotation Forest: A New Classifier Ensemble Method by Juan J. Rodrı´guez, Ludmila I. Kuncheva and Carlos J. Alonso
- ➔ Machine learning applications in cancer prognosis and prediction by Konstantina Kourou, Themis P. Exarchos, Konstantinos P. Exarchos, Michalis V. Karamouzis, Dimitrios I. Fotiadis

THANK YOU