Lecture 2 - 07-04-2020

1.1 Argomento

Classification tasks
Semantic label space Y
Categorization Y finite and
small Regression Y appartiene ad |R
How to predict labels?
Using the lost function → ..
Binary classification
Label space is Y = -1, +1
Zero-one loss

$$\ell(y, \hat{y} = \begin{cases} 0, & \text{if } \hat{y} = y \\ 1, & \text{if } \hat{y} \neq y \end{cases}$$

$FP \quad \hat{y} = 1, \quad y = -1$
$FN \quad \hat{y} = -1, \quad y = 1$

Losses for regression?
y, and $\hat{y} \in \mathbb{R}$,
so they are numbers!
One example of loss is the absolute loss: absolute difference between numbers

1.2 Loss

1.2.1 Absolute Loss

$$\ell(y, \hat{y} = |y - \hat{y}| \Rightarrow absolute \quad loss$$

Some inconvenient properties:

- ...
- Derivative only two values (not much informations)

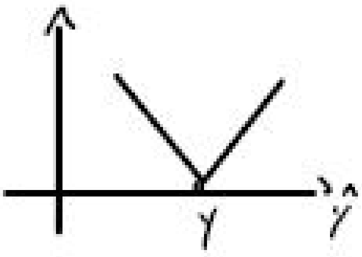

Figure 1.1: Example of domain of K_{NN}

1.2.2 Square Loss

$$\ell(y, \hat{y} = (y - \hat{y})^2 \Rightarrow \textit{square loss}$$

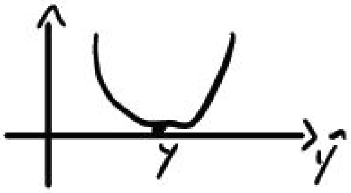

Figure 1.2: Example of domain of K_{NN}

Derivative :

- more informative
- and differentible

Real numbers as label → regression.
Whenever taking difference between two prediction make sense (value are numbers) then we are talking about regression problem.
Classification as categorization when we have small finite set.

1.2.3 Example of information of square loss

$\ell(y, \hat{y}) = (y - \hat{y})^2 = F(y)$
$F'(\hat{y}) = -2 \cdot (y - \hat{y})$

- I'm under sho or over and how much
- How much far away from the truth

$\ell(y, \hat{y}) = |y - \hat{y}| = F(y') \cdot F'(y) = Sign(y - \hat{y})$

Question about the future
Will it rain tomorrow?
We have a label and this is a binary classification problem.
My label space will be Y = "rain", "no rain"
We don't get a binary prediction, we need another space called prediction space (or decision space).
$Z = [0, 1]$
$\hat{y} \in Z \qquad \hat{y}$ *is my prediction of rain tomorrow*
$\hat{y} = \mathbb{P}(y = "rain") \quad \rightarrow$ *my guess is tomorrow will rain (not sure)*

$y \in Y \qquad \hat{y} \in Z$
quadHow can we manage loss?
Put numbers in our space
$\{1, 0\}$ *where 1 is rain and 0 no rain*

I measure how much I'm far from reality.
So loss behave like this and the punishment is gonna go linearly??

26..

However is pretty annoying. Sometime I prefer to punish more so i going quadratically instead of linearly.
There are other way to punish this.
I called **logarithmic loss**
We are extending a lot the range of our loss function.

$$\ell(y, \hat{y}) = |y - \hat{y}| \in |0, 1| \qquad \ell(y, \hat{y}) = (y - \hat{y})^2 \in |0, 1|$$

If i want to expand the punishment i use logarithmic loss

$$\ell(y, \hat{y} = \begin{cases} ln\frac{1}{\hat{y}}, & \text{if } y = 1 (rain) \\ ln\frac{1}{1-\hat{y}}, & \text{if } y = 0 (no\ rain \end{cases}$$

$F(\hat{y}) \rightarrow can\ be\ 0\ if\ i\ predict\ with\ certainty$
$If \quad \hat{y} = 0.5 \qquad \ell(y, \frac{1}{2}) = ln2 \quad constant\ losses\ in\ each\ prediction$

$\lim_{\hat{y} \to 0^+} \ell(1, \hat{y}) = +\infty$
We give a vanishing probability not rain but tomorrow will rain.
So this is $+\infty$
$\lim_{\hat{y} \to 1^-} \ell(0, \hat{y}) = +\infty$

The algorithm will be punish high more the prediction is not real. Algorithm will not get 0 and 1 because for example is impossible to get a perfect prediction.
This loss is useful to give this information to the algorithm.

Now we talk about labels and losses

1.2.4 labels and losses

Data points: they have some semantic labels that denote some true about this data points and we want to predict this labels.
We need to define what data points are: number? Strings? File? Typically they are stored in database records
They can have very precise structure or more homogeneously structured
A data point can be viewed as a vector in some d dimensional real space. So it's a vector of number

$$\mathbb{R}^d X = (x_1, x_2..., x_d) \in \mathbb{R}^c$$

Image can be viewed as a vector of pixel values (grey scale 0-255).
I can use geometry to learn because point are in my Euclidean space. Data can be represented as point in Euclidean space. Images are list of pixel that are pretty much the same range and structure (from 0 to 255). It's very natural to put them in a space.

Assume X can be a record with heterogeneous fields:
For example medical records, we have several values and each fields has his meaning by it's own. (Sex, weight, height, age, zip code)

Each one has a different range, in some cases is numerical but something have like age ..
Does have any sense to see a medical record as a point since coordinates have different meaning.
Fields are not comparable.
This is something that you do: when you want to solve some inference you have to decide which are the label and what is the label space and we have to encode the data points.

Data algorithm expect some homogenous interface. In this case algorithm has to build records with different values of fields.
This is something that we have to pay attention too.
You can always each range of values in number. So ages is number, sex you can give 0 and 1, weight number and zip code is number.
How ever geometry doesn't make sense since I cannot compare this coordinates.
Linear space i can sum up as vector: i can make linear combination of vectors.
Inner product to measure angles! (We will see in linear classifier).

I can scramble the number of my zip code.
So we get problems with sex and zip code

Why do we care about geometry? I can use geometry to learn.
However there is more to that, geometry will carry some semantically information that I'm going to preserve during prediction.
I want to encode my images as vectors in a space. Images with dog.....

PCA doesn't work because assume we encode in linear space.
We hope geometry will help us to predict label correctly and sometimes i hard to convert data into geometry point.
Example of comparable data: images, or documents.
Assume we have documents with corpus (set of documents).
Maybe in English and talk about different thing and different words.
X is a document and i want to encode X into a point fix in bidimensional space.
There is a way to encode a set of documents in point in a fixed dimensional space in such way it make sense this coordinate are comparable.
I can represent fields with [0,1] for Neural network for example. But they have no geometrical meaning

1.2.5 Example TF(idf) documents encoding

TF encoding of docs.

1. Extract where all the words from docs
2. Normalize words (nouns, adjectives, verbs ...)
3. Build a dictionary of normalized words

Doc $x = (x_1, .., x_d)$
I associate a coordinate for each word in a dictionary.
d = number of words in dictionary
I can decide that
$x_i = 1$ *If i-th word of dictionary occurs in doc.*
$x_i = 0$ *Else*

X_i *number of time i-th word occur in doc.*
Longer documents will have higher value of coordinates that are not zero.
Now i can do the TF encoding in which xi = frequency with which i-th word occur in dictionary.
You cannot sum dog and cat but we are considering them frequencies so we are summing frequency of words.
This encoding works well in real words.
I can choose different way of encoding my data and sometime i can encode a real vector

I want

1. A predictor $f : X \longrightarrow Y$ (in weather $X \longrightarrow Z$
2. X is our data space (where points live)
3. $X = \mathbb{R}^d$ images
4. $X = X_1 x ... x X_d$ Medical record
5. $\hat{y} = f(x)$ predictor for X

(x, y)

We want to predict a label that is much closer to our label. How?
Loss function: so this is my setting and is called and example.
Data point together with label is a "example"

We can get collection of example making measurements or asking people. So we can always recover the true label.
We want to replace this process with a predictor (so we don't have to bored a person).
y is the ground truth for x → mean reality!
If i want to predict stock for tomorrow, i will wait tomorrow to see the ground truth.