سوالات متنی:

یک.

الف) پیوسته، کمی-نسبت **ب)** پیوسته، کمی-بازه (بازه اعداد حقیقی به متر) **پ)** پیوسته، کمی-بازه
ت) دودویی، کیفی-ترتیبی (با فرض طلا > نقره > برنز) **ث)** پیوسته، کمی-بازه. **ج)** گسسته، کیفی-ترتیبی (خیلی کم، کم، متوسط، زیاد، خیلی زیاد)

دو.

الف) نویز در میان training data خوب نیست، چون تابع را از هدف اصلی دور می‌کند، اما post-training و تنها در هنگام محافظت از adversarial attacks روی مدل (در مورد شبکه‌های عصبی به طور خاص) به کار می‌آید. اما outlier مطلوب است چرا که هم باعث جلوگیری از overfitting می‌شود و تابع ما را به تابع هدف نزدیک‌تر می‌کند.
ب) نویز می‌تواند دیتا را بی‌قاعده‌تر یا غیرمعمول‌تر نشان دهد، به همین علت ممکن است بعضی داده‌ها به شکل outlier به نظر بیایند.
پ) نویز می‌تواند نزدیک به دیتای عادی باشد، پس همیشه به صورت outlier نخواهد بود.
ت) خیر. چون مشخصه‌ی تمایز outlier متفاوت بودن از دیتای اصلی و مشخصه‌ی تمایز noise بی‌ربط بودن به تابع هدف است.

سه.

$$Cos(x,y) = \frac{x.y}{\sqrt{(x.x)(y.y)}} \quad , Correlation: \rho_{x,y} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \sum_{i=1}^{n}(y_i - \bar{y})^2}},$$

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

a) Mean(x) = 0, Mean(y) = 0, Cosine = 0, Euclidean: 2.0, Correlation: 0
b) Mean(x) = 0, Mean(y) = $-\frac{1}{3}$, Cosine = 0, Euclidean: 4.69, Correlation: 0
c) Mean(x) = $\frac{2}{3}$, Mean(y) = $\frac{2}{3}$, Cosine: 0.75, Jaccard: 0.4, Correlation: $\frac{1}{4}$

چهار.

الف) یک. حذف کردن تاپل‌های ناقص – دو. جای‌گزین کردن مقدارهای ناموجود تاپل‌های ناقص با میانگین همان مقدار در بقیه تاپل‌ها.
ب) یک. حذف کردن فیچرهایی که واریانس پایینی دارند (یادگیری آن‌ها چیز خاصی به مدل اضافه نمی‌کند). دو. PCA: به این صورت که در فضای n بعدی داده‌ها، با پیدا کردن صفحه‌ی n-1بعدی‌ای که بیش‌ترین تعداد داده‌ها روی آن صفحه قرار دارند، با تغییر بردارهای پایه به فضای n-1 بعدی این صفحه، یک بعد (یا با تکرار این فرآیند، حتی ابعاد بیش‌تر) از داده‌ها حذف می‌شوند.
پ) Oversampling که به معنای کپی‌کردن برخی داده‌های کم‌تعدادتر است (بایاس را زیاد می‌کند) و Undersampling که به معنای پاک‌کردن تعدادی از داده‌های پرتعدادتر است (که می‌تواند واریانس را زیاد کند)
ت) Hold-Out: درصد بیش‌تری از دیتاست را به test اختصاص دهیم و Cross-Validation که به معنای تقسیم دیتا به k قسمت

مختلف است و با انجام kبار iteration کلی، هر بار ۱ عدد از قسمت‌ها را به عنوان test و بقیه را به عنوان training در نظر می‌گیریم که حتما تمامی دیتا در فاز training استفاده شده باشد.

ث) یک. بیش‌تر کردن تعداد پارامترها و افزایش پیچیدگی مدل. دو. زیاد کردن Training Time تا cost function به مقدار کافی minimize شود.

پنج.

الف) بله. اگر تعداد outlierها از تعداد مشخصی بیش‌تر باشد، به احتمال زیاد شیب را به صورتی تغییر می‌دهد که پیش‌بینی مدل برای تعداد زیادی از داده‌ها غلط می‌شود.

ب) $J = \frac{1}{n}\sum_{i=1}^{n}(pred_i - y_i)^2$ است که به عبارتی میانگین توان۲ ی خطاهاست. توان ۲ به این علت حضور دارد که برخی خطاها مثبت و بعضی منفی هستند و می‌خواهیم همه را به یک دید ببینیم. (و به این علت که قدر مطلق محاسبات را پیچیده می‌کند)

پ)

$X^T X\beta = X^T y \rightarrow \beta = (X^T X)^{-1} X^T y,$
Appending a column full of ones as x^0 to the beginning of X, we get:
$\beta = [-41.887, 62.66]$

شش.

Shannon Entropy: $-\Sigma_i P(x)\log_2(P(x))$

الف)

$P(X = weak) = 0.35\ , P(X = average) = 0.275\ , P(X = rich) = 0.375$
$H(X) = 1.573$

$P(Y = Democrat) = 0.5$, P(Y=Republican) = 0.5
H(Y) = 1.0

ب)

		حزب	
		دموکرات	ج.خواه
	ضعیف	0.175	0.175
طبقه	متوسط	0.1375	0.1375
	مرفه	0.1875	0.1875

$P(P_{arty}) * P(C_{lass}) =$

			حزب
		دموکرات	ج.خواه
	ضعیف	0.1	0.25
طبقه	متوسط	0.175	0.1
	مرفه	0.225	0.15

$P(P_{arty}, C_{lass}) =$

$I(P_{arty}, C_{lass}) = \Sigma_{P_{arty}}\Sigma_{C_{lass}} P(P_{arty}, C_{lass}) \log_2 \frac{P(P_{arty}, C_{lass})}{P(P_{arty})P(C_{lass})} = 0.074$

پ) چون میزان Mutual Information پایین است این دو متغیر مستقل نیستند اما دانستن یکی از آن‌ها قدرت پیش‌بینی زیادی برای دیگری نمی‌دهد.

هفت.

الف) Aggregation: تبدیل چند ویژگی/داده به یک ویژگی/داده. با کم‌کردن ویژگی‌ها/داده‌ها، میزان فضای ذخیره‌سازی و پردازش را بهینه‌سازی می‌کنیم.

Sampling: قسمت کوچک‌تری از دیتا که دارای همان خصوصیات دیتای اصلی است را به عنوان Sample انتخاب می‌کنیم، یک ملاک خوب برای انتخاب یک Sample مناسب دارا بودن میانگین یکسان با داده‌ی اصلی است.

Dimensionality Reduction: ابعاد داده‌ها را کم می‌کنیم، هم فضای ذخیره‌سازی و زمان پردازش بهینه می‌شوند، هم می‌توانیم بهتر داده را visualize کنیم و هم با تکنیک‌های موجود می‌توانیم ویژگی‌های بی‌اهمیت و نویز را کم کنیم.

ب)

Min = 200, Max = 1000. [200 => 0, 300 => $\frac{1}{8}$, 400 => 0.25, 600 => 0.5, 1000 => 1]

μ: 500, σ: 316.23, Z-Score(x_i) = $\frac{x_i-\mu}{\sigma}$ => [-0.9487, -0.6324, -0.3162, 0.3162, 1.5811]

هشت.

الف)

$$Cost(\beta) = \alpha\,|\beta|^2 + \,|X\beta - y|^2 = \lambda\beta^T\beta + (X\beta - y)^T(X\beta - y)$$
$$Minimizing, we\ set \frac{\partial Cost(\beta)}{\partial(\beta)} = 2\lambda\beta + 2X^T(X\beta - y) = 0 \rightarrow \lambda\beta + X^T(X\beta - y) = 0$$
$$= (\lambda I + X^TX)\beta - X^Ty = 0 \rightarrow (\lambda I + X^TX)\beta = X^Ty \rightarrow \beta = \,(\lambda I + X^TX)^{-1}X^Ty$$

ب)

Writing the cost function in terms of sums instead of matrices we have:

Cost = $\Sigma_{i=1}^{N}\left(y_i - \Sigma_{j=0}^{M}\beta_j x_{ij}\right)^2 + \lambda\Sigma_{j=0}^{M}w_j^2$

$$\frac{\partial Cost}{\partial \beta_j} = \,-2\Sigma_{i=1}^{N}x_{ij}\left(y_i - \Sigma_{k=0}^{M}\beta_k x_{ik}\right) + 2\lambda\beta_j$$

In gradient descent, each iteration is updated using the previous weight and the learning speed η:

$$\beta_j^{t+1} = \beta_j^t - \eta\left[-2\Sigma_{i=1}^{N}x_{ij}\left(y_i - \Sigma_{k=0}^{M}\beta_k * x_{ij}\right) + 2\lambda\beta_j\right] \rightarrow$$
$$\beta_j^{t+1} = (1 - 2\lambda\eta)\beta_j^t + 2\eta\Sigma_{i=1}^{N}x_{ij}\left(y_i - \Sigma_{k=0}^{M}\beta_k * x_{ik}\right) \rightarrow$$

Ridge regression is equivalent to reducing the weight β_j by a factor of $(1 - 2\lambda\eta)$ and then applying the same update rule used by the ordinary least squares method.

پ)

Assuming $g(\beta) = \lambda|\beta|^2$ and knowing that given the definition of a convex function, if $\lambda \geq 0$, then the function is convex

$$g(\phi\beta_1 + (1-\phi)\beta_2) = \lambda|\phi\beta_1 + (1-\phi)\beta_2|^2 \leq \lambda(\phi|\beta_1|^2 + (1 - \phi|\beta_2|^2) = \phi g(\beta_1) + (1-\phi)g(\beta_2)$$
$$\rightarrow\ \lambda|\phi\beta_1 + (1-\phi)\beta_2|^2 \leq\ \phi g(\beta_1) + (1-\phi)g(\beta_2)$$

گزارش پیاده‌سازی:

قسمت اول:

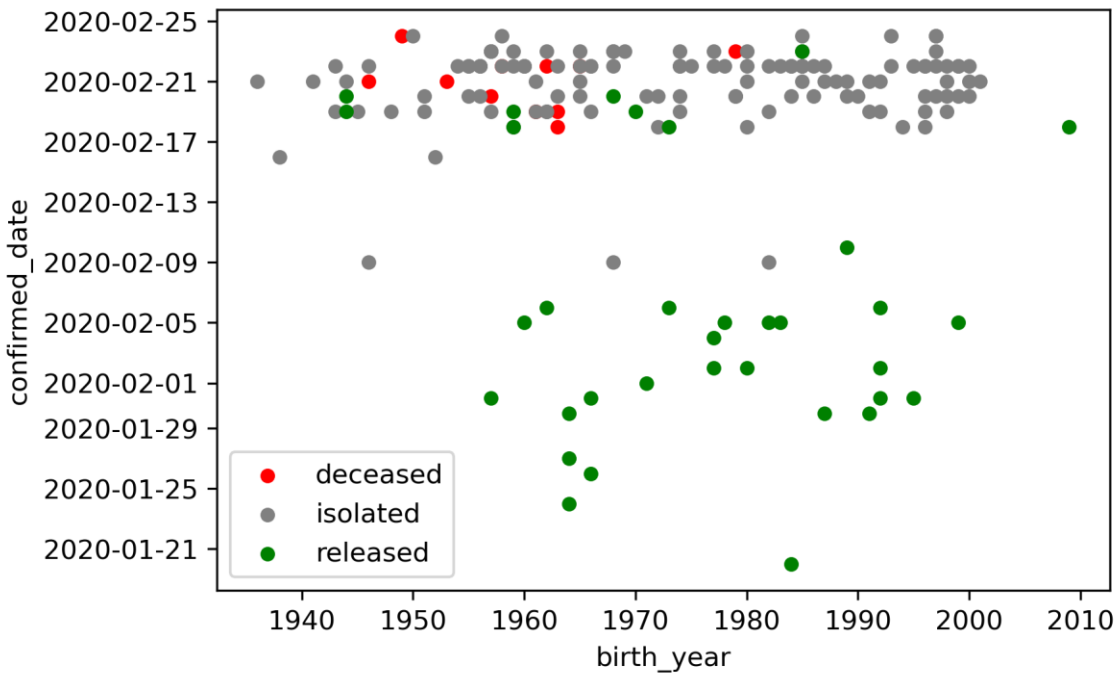
یک.

	id	sex	birth_year	country	region	infection_reason	infected_by	confirmed_date	state
0	1	female	1984.0	China	filtered at airport	visit to Wuhan	NaN	1/20/2020	released
1	2	male	1964.0	Korea	filtered at airport	visit to Wuhan	NaN	1/24/2020	released
2	3	male	1966.0	Korea	capital area	visit to Wuhan	NaN	1/26/2020	released
3	4	male	1964.0	Korea	capital area	visit to Wuhan	NaN	1/27/2020	released
4	5	male	1987.0	Korea	capital area	visit to Wuhan	NaN	1/30/2020	released
...	...	...	...	...	...	...	...	...	...
171	172	female	1997.0	Korea	Gyeongsangbuk-do	NaN	NaN	2/24/2020	isolated
172	173	male	1949.0	Korea	Daegu	NaN	NaN	2/24/2020	deceased
173	174	female	1958.0	Korea	Gyeongsangbuk-do	NaN	NaN	2/24/2020	isolated
174	175	male	1997.0	Korea	Gyeongsangbuk-do	NaN	NaN	2/24/2020	isolated
175	176	female	1950.0	Korea	capital area	NaN	NaN	2/24/2020	isolated

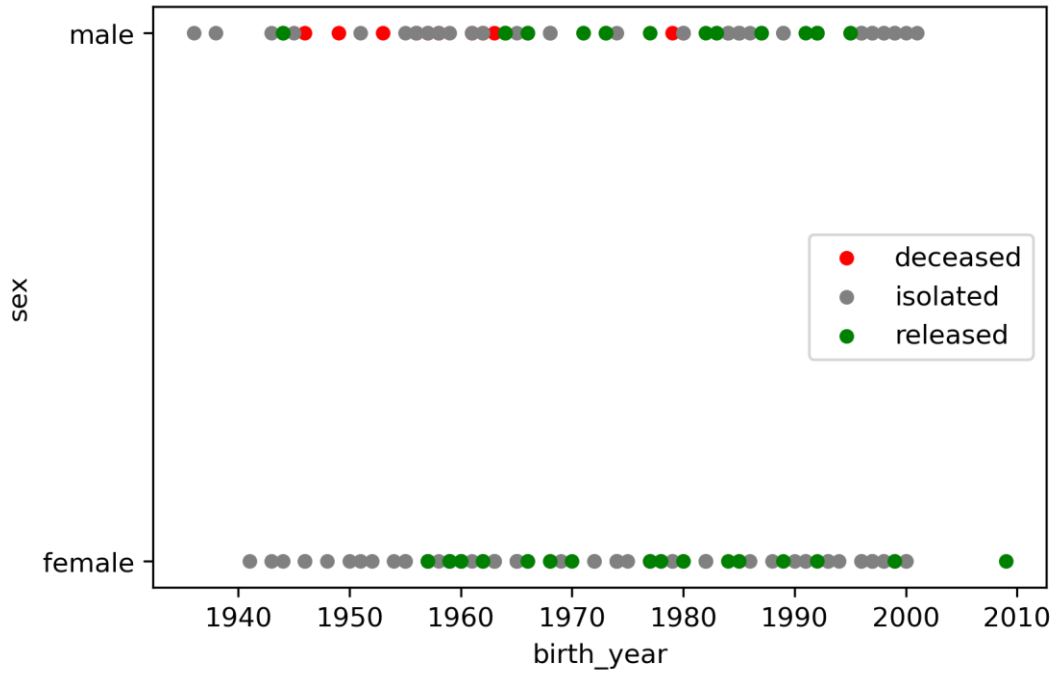
دو. داده‌های مبتلایان کرونا در کشورهای چین و کره را نشان می‌دهد که جنسیت، کشور، سال تولد، محل زندگی، دلیل ابتلا، فرد مبتلا کننده، تاریخ تایید ابتدا و وضعیت بیمار را نشان می‌دهد.

سه.

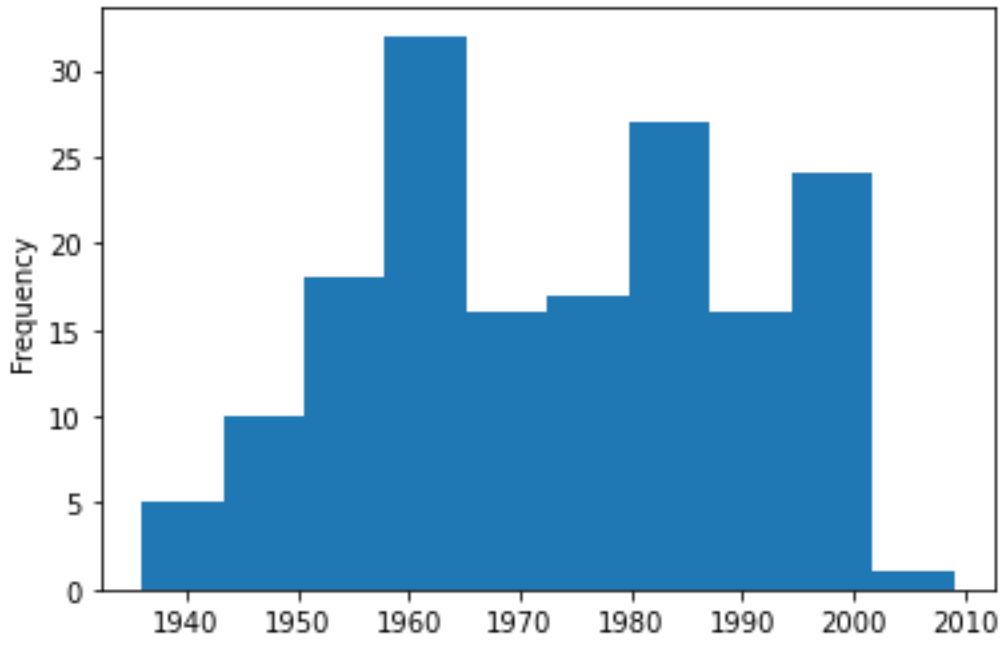
Scatter Plot (by date):



Scatter Plot (by sex):



Histogram Plot (by birth year)



به نظر می‌رسد که در نمودار Scatter by date بیمارهایی که قبل از 2020-02-09 وضعیت‌شان تایید شده و هنوز به وضعیت release نرسیده‌اند را می‌توانیم به عنوان outlier در نظر بگیریم و وضعیت آن‌ها را به released تغییر دهیم. حتی می‌توان موردی که حدود سال ۱۹۸۰ متولد شده و فوت شده است را نیز به عنوان outlier در نظر بگیریم و دیتای آن را حذف کنیم.

قسمت دوم:

با فرض این‌که خطاهای کمتر از یک نمره قابل قبول است، میزان دقت مدل 69.62 درصد محاسبه می‌شود.