به نام خدا

پاسخ تمرین اول

یادگیری ماشین

علی عدالت ۸۱۰۱۹۹۳۴۸

۱)

الف) تعداد کل داده‌ها ۲۰۰ تا است که کم محسوب می‌شود و به همین دلیل در حالت اول تعداد ۱۰۰ داده برای آموزش کافی نیست. به دلیل کمبود داده‌های آموزش[1] عملیات یادگیری به طور کامل انجام نشده است. به دلیل کامل نبودن یادگیری، عملکرد طبقه‌بند بهینه نیست. دلیل افزایش دقت با افزایش داده‌های آموزش، یادگیری بهتر است که با دیدن داده‌های بیشتر بدست آمده است. در حالت دوم تعداد داده‌های آموزش مناسب است ولی تعداد داه‌های تست نسبتا کم است. در این حالت تعداد زیاد داده‌ی آموزش باعث یادگیری مناسب شده است و به نظر می‌رسد عملکرد بهینه بوده است ولی در این جا چون تعداد کل داده‌ها و تعداد داده تست کم است با یک بار انجام طبقه بندی و یک دقت نمی‌توان نتیجه قطعی گرفت. دلیل این موضوع این است که می‌تواند نتیجه حاصل به خاطر این دسته بندی خاص داده‌های تست و آموزش باشد. در این جا با استفاده از k fold و تست کردن با همین تعداد می‌توان قطعی صحبت کرد.

ب) فرمول به شکل زیر است. تخمین وزن‌ها و بایاس‌ها در مسئله رگرسیون خطی را می‌توان با استنباط آماری مورد بررسی قرار داد. از این استنباط می‌توان برای تخمین توزیع احتمال یک رخداد یا آماره‌هایی از آن استفاده کرد. برای مثال تخمین توزیع احتمال تعلق نمونه به یک دسته در طبقه بندی. برای آزمایش فرضیه[2] و پاسخ به سوالاتی در باره‌ی توزیع احتمال می‌توان از آن استفاده کرد. برای مثال آیا احتمال رخداد پدیده از نیم بیشتر است؟

---

[1] train

[2] Hypothesis testing

$$\text{Statistical inference} = \text{Probability}^{-1}$$

۲)

الف) در این الگوریتم ما به دنبال θ ای هستیم که تابع هزینه را کمینه کند. در این الگوریتم ابتدا یک θ اولیه دلخواه انتخاب می‌کنیم و سپس به طور مرتب مقدار درایه‌های θ را به گونه‌ای به روز می‌کنیم تا مقدار تابع هزینه کمتر شود. امیدواریم به مقداری از θ همگرا شویم که تابع هزینه را به حداقل برساند. فرمول به روز رسانی θ به صورت زیر است.

$$\theta_j := \theta_j - \alpha \,.\, \frac{\partial J(\theta)}{\partial \theta_j}$$

در این جا θ شامل w و b است.

ب) در این جا w و b را یک بعدی در نظر می‌گیریم. فرمول به روز رسانی w برای یک زوج داده (x_i, y_i) را در زیر محاسبه می‌کنیم.

$$w := w - \alpha \,.\, \frac{\partial J(\theta)}{\partial w}$$

$$\frac{\partial J(\theta)}{\partial w} = \frac{\partial}{\partial w}\left(\frac{1}{2} \,.\, (ln(1+e^{w.x+b}) - y)^2\right)$$

$$= (ln(1+e^{w.x+b}) - y) \,.\, \frac{\partial}{\partial w}(\, ln(1+e^{w.x+b}) - y\,)$$

$$= (ln(1+e^{w.x+b}) - y) \,.\, \left(\frac{x \,.\, e^{(w.x+b)}}{1+e^{(w.x+b)}}\right)$$

$$w := w - \alpha \,.\, (ln(1+e^{wx_i+b}) - y_i) \,.\, \left(\frac{x_i \,.\, e^{(w.x_i+b)}}{1+e^{(w.x_i+b)}}\right)$$

فرمول به روز رسانی b برای یک زوج داده (x_i, y_i) را در زیر محاسبه می‌کنیم.

$$b := b - \alpha \,.\, \frac{\partial J(\theta)}{\partial b}$$

$$\frac{\partial J(\theta)}{\partial b} = \frac{\partial}{\partial b}\left(\frac{1}{2} \,.\, (ln(1+e^{w.x+b}) - y)^2\right)$$

$$= (ln(1+e^{w.x+b}) - y) \,.\, \frac{\partial}{\partial b}(\, ln(1+e^{w.x+b}) - y\,)$$

$$= (ln(1 + e^{w.x+b}) - y) . (\frac{e^{(w.x+b)}}{1 + e^{(w.x+b)}})$$

$$b := b - \alpha . (ln(1 + e^{w.x_i+b}) - y_i) . (\frac{e^{(w.x_i+b)}}{1 + e^{(w.x_i+b)}})$$

نرخ یادگیری تعیین می‌کند که گامی که در جهت گرادیان بر می داریم چقدر بزرگ باشد. اگر این نرخ خیلی بزرگ باشد، ما اطراف مینیمم پرش می‌کنیم و به آن نمی‌رسیم. اگر خیلی بزرگ باشد، زمان رسیدن به مینیمم بسیار طولانی می‌شود یا باعث می‌شود در مینیمم محلی نامطلوب گیر کنیم.

۳) چند جمله‌ای مورد نظر به صورت زیر است.

$$L(x) := \sum_{j=0}^{k} y_j \ell_j(x)$$

$$\ell_j(x) := \prod_{\substack{0 \le m \le k \\ m \ne j}} \frac{x - x_m}{x_j - x_m} = \frac{(x - x_0)}{(x_j - x_0)} \cdots \frac{(x - x_{j-1})}{(x_j - x_{j-1})} \frac{(x - x_{j+1})}{(x_j - x_{j+1})} \cdots \frac{(x - x_k)}{(x_j - x_k)}$$

با توجه به این فرض اولیه که هیچ دو نقطه x یکسانی ندارند پس برای $m \neq j$ داریم $x_j - x_m \neq 0$ پس این عبارت همیشه خوش تعریف است. تابع فقط می تواند برای هر آرگومان x_i یک مقدار خروجی داشته باشد. به همین دلیل اگر دو نقطه x برابری داشته باشند حتما y برابری دارند. در تابع بالا به ازای هر $i \neq j$ عبارت زیر برقرار است.

$$\forall (j \neq i) : \ell_j(x_i) = \prod_{m \neq j} \frac{x_i - x_m}{x_j - x_m} = \frac{(x_i - x_0)}{(x_j - x_0)} \cdots \frac{(x_i - x_i)}{(x_j - x_i)} \cdots \frac{(x_i - x_k)}{(x_j - x_k)} = 0$$

و برای $i = j$ عبارت زیر برقرار است.

$$\ell_j(x_j) := \prod_{m \neq j} \frac{x_j - x_m}{x_j - x_m} = 1$$

بنابراین مقدار تابع L در هر نقطه با طول x_j برابر y_j است که به صورت زیر محاسبه می‌شود. پس تابع L از تمام نقاط می‌گذرد.

$$L(x_j) = y_j + 0 + 0 + \cdots + 0 = y_j$$

۴)

$$cov(\beta_1,\ \beta_0) = E[E[y].\beta_1 - \beta_1^2.E[x]] - E[\beta_1].E[E[y] - \beta_1.E[x]]$$
$$= E[E[y].\beta_1] - E[\beta_1^2.E[x]] - E[E[y]].E[\beta_1] + E[\beta_1.E[x]].E[\beta_1]$$
$$= \ -E[x].var(\beta_1)$$

برای استقلال دو پارامتر، باید کوواریانس آن‌ها صفر باشد که دلیل آن توزیع گوسی پارامترها است. برای صفر شدن کوواریانس یا باید میانگین x ها صفر باشد یا واریانس β_1 . با استفاده از ماتریس variance-covariance داریم:

$$Var(\hat{\beta}) := \sigma^2(\hat{\beta}) = \begin{pmatrix} Var(\hat{\beta}_0) & Cov(\hat{\beta}_0, \hat{\beta}_1) \\ Cov(\hat{\beta}_0, \hat{\beta}_1) & Var(\hat{\beta}_1) \end{pmatrix}$$

از طرفی بر اساس تعریف واریانس داریم:

$$\begin{aligned} Var(\hat{\beta}) &= E[\hat{\beta}^2] - E[\hat{\beta}]^2 = E[((X'X)^{-1}X'Y)^2] - \beta^2 = E[((X'X)^{-1}X'(X\beta + u))^2] - \beta^2 = \\ &= E[((X'X)^{-1}X'X\beta + (X'X)^{-1}X'u))^2] - \beta^2 = E[(\beta + (X'X)^{-1}X'u))^2] - \beta^2 = \\ &= E[\beta^2] + 2(X'X)^{-1}X'E[u] + E[((X'X)^{-1}X'u))^2] - \beta^2 = \\ &= \beta^2 + 0 + E[((X'X)^{-1}X'u))^2] - \beta^2 = E[((X'X)^{-1}X'u))^2] = \\ &= \left((X'X)^{-1}X'\right)^2 \cdot E[u^2] \end{aligned}$$

$$E[u^2] = Var(u) = \sigma^2$$

$$\begin{aligned} Var(\hat{\beta}) &= \left((X'X)^{-1}X'\right)^2 \cdot E[u^2] = (X'X)^{-1}X' \cdot (X'X)^{-1}X' \cdot \sigma^2 = \sigma^2(X'X)^{-1} \cdot I = \\ &= \sigma^2(X'X)^{-1}. \end{aligned}$$

بر اساس عبارت بالا واریانس بتاها و کوواریانس آن‌ها به صورت زیر است. برای صفر شدن کوواریانس بر اساس فرمول نهایی باید میانگین x ها صفر باشد.

$$(X'X)^{-1} = \begin{pmatrix} \frac{\sum x_i^2}{n\sum(x_1-\bar{x})^2} & \frac{-\sum x_i}{n\sum(x_1-\bar{x})^2} \\ \frac{-\sum x_i}{n\sum(x_1-\bar{x})^2} & \frac{1}{\sum(x_1-\bar{x})^2} \end{pmatrix}$$

$$Var(\hat{\beta}) = \begin{pmatrix} Var(\hat{\beta}_0) & Cov(\hat{\beta}_0, \hat{\beta}_1) \\ Cov(\hat{\beta}_0, \hat{\beta}_1 & Var(\hat{\beta}_1) \end{pmatrix} = \sigma^2(X'X)^{-1} =$$

$$= \begin{pmatrix} \frac{\sigma^2\sum x_i^2}{n\sum(x_1-\bar{x})^2} & \frac{-\sigma^2\sum x_i}{n\sum(x_1-\bar{x})^2} \\ \frac{-\sigma^2\sum x_i}{n\sum(x_1-\bar{x})^2} & \frac{\sigma^2}{\sum(x_1-\bar{x})^2} \end{pmatrix}$$

۵)

الف) فرمول β_1 ، β_2 و δ^2 در زیر آمده است. بر اساس این فرمول‌ها ما نیاز به محاسبه‌ی $E[x]$ ، $E[y]$ ، $E[xy]$ و $E[x^2]$ برای محاسبه‌ی β_1 و β_2 داریم.

$$\hat{\sigma}^2 = s^2 = \frac{SSE}{n-2} = \frac{\sum(y-\hat{y}_i)^2}{n-2}$$

$$\beta_1 = \frac{\sum_{i=1}^{n}(Y_i-\bar{y})(x_i-\bar{x})}{\sum_{i=1}^{n}(x_i-\bar{x})^2}, \beta_0 = \bar{Y} - \beta_1\bar{x}$$

$$\beta_1 = \frac{cov(x,y)}{var(x)}$$

در ادامه به محاسبه‌ی متغییرها می‌پردازیم.

$$E[x] = \frac{(4+9+10+14+4+7+12+22+1+17)}{10} = 10$$

$$E[x^2] = 137.6 \ , \ E[xy] = 694.5 \ , \ E[y] = 56.4$$

$$\beta_1 = \frac{E[xy]-E[x]E[y]}{E[x^2]-E[x]E[x]} = \frac{694.5-(10\times 56.4)}{137.6-(100)} = 3.47074468085$$

$$\beta_0 = 56.4-(3.47074468085\times 10) = 21.6925531915$$

$$\sigma^2 = 27.8847739362$$

برای محاسبه‌ی واریانس بتاها از فرمول‌های زیر استفاده می‌کنیم.

$$\mathrm{Var}\left(\hat{\beta}_0\right) = \frac{\sigma^2 \sum_{i=1}^{n} x_i^2}{n \sum_{i=1}^{n} (x_i - \bar{x})^2}$$

$$\mathrm{Var}\left(\hat{\beta}_1\right) = \frac{\sigma^2}{\sum_{i=1}^{n} (x_i - \bar{x})^2}$$

$$var(\beta_1) = \frac{27.8847739362}{376} = 0.074161632809$$

$$var(\beta_0) = var(\beta_1)\times E[x^2] = 10.2046406745$$

ب) برای محاسبه‌ی کوواریانس و کرولیشن از فرمول‌های زیر استفاده می‌کنیم.

$$\mathrm{cov}\left(\hat{\beta}_0, \hat{\beta}_1\right) = \frac{-\sigma^2 \sum_{i=1}^{n} x_i}{n \sum_{i=1}^{n} (x_i - \bar{x})^2}$$

$$corr(\beta_0, \beta_1) = \frac{cov(\beta_0, \beta_1)}{\sqrt{var(\beta_0)\times var(\beta_1)}}$$

$$cov(\beta_0, \beta_1) = -E[x]\times var(\beta_1) = -0.74161632809$$

$$corr(\beta_0, \beta_1) = -0.85249292433$$

۶) در داده‌ها تعدادی عکس وجود دارد که لیبل آن‌ها از روی نام فایل مشخص نیست. نام این عکس‌ها با توجه به تیم درون عکس اصلاح کردیم که در زیر آمده است.

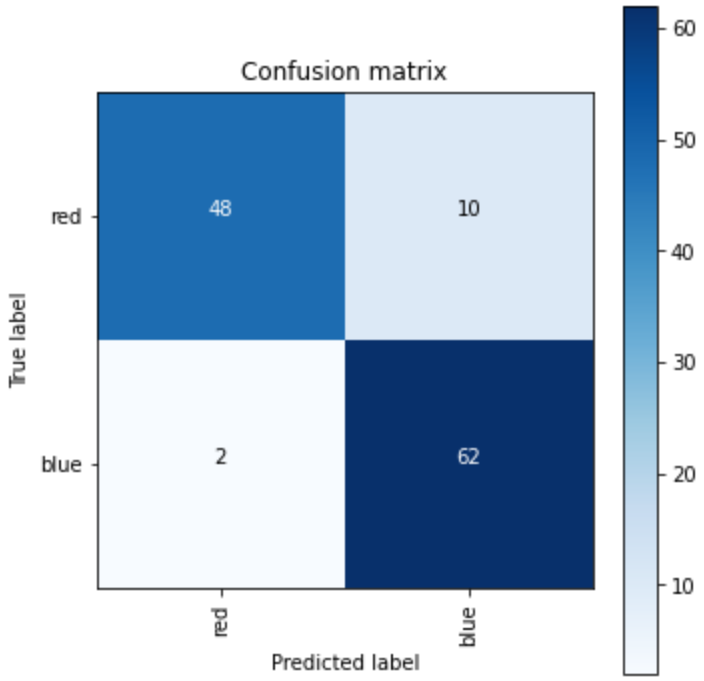
```
rename_map = {'1.jpg' : 'm66.jpg',
              'ManUtd-50887805157607364174 3_medium.jpg' : 'm65.jpg',
              'images.jpg' : 'm64.jpg',
              'index.jpg' : 'c66.jpg',
              }
```

لیبل هر عکس از روی حرف اول نام فایل که m یا c است به ترتیب 0 و 1 در نظر می‌گیریم. برای تشخیص کلاس هر عکس، میانگین تمام رنگ‌های پیکسل‌های عکس را با رنگ آبی(چلسی) و قرمز(یونایتد) مقایسه کردم. برای این کار مقدار red در تمام پیکسل ها میانگین گرفته می‌شود و به عنوان ویژگی میزان قرمزی عکس درنظر گرفته می‌شود. برای green و blue در پیکسل‌ها نیز همین کار را می‌کنیم تا ویژگی های میزان سبزی و میزان آبی بودن برای یک عکس بدست بیاید. رنگ آبی(چلسی) میانگین RGB رنگ‌هایی است که آبی نامیده می‌شوند که این رنگ‌ها از لینک اول[3] بدست آمده است. رنگ قرمز(یونایتد) میانگین RGB رنگ‌هایی است که قرمز نامیده می‌شوند که این رنگ‌ها از لینک دوم[4] بدست آمده است. برای تعیین لیبل، توان دوم فاصله اقلیدسی ویژگی‌های عکس از رنگ قرمز(یونایتد) و رنگ آبی(چلسی) را استفاده کردم. اگر فاصله ویژگی‌های عکس از رنگ آبی(چلسی) کمتر باشد لیبل آبی یا 1 می‌گیرد و اگر فاصله ویژگی‌های عکس از رنگ قرمز(یونایتد) کمتر باشد لیبل قرمز یا 0 می‌گیرد. سپس موارد خواسته شده محاسبه شده است. مقدار correct classification برابر حاصل تقسیم مجموع اعداد روی قطر اصلی ماتریس confusion بر تعداد کل داده‌ها است. تعداد کل داده‌ها برابر مجموع تمام درایه‌های ماتریس confusion است. این مقدار با مقدار accuracy برابر است که نشان دهنده‌ی درصد داده‌هایی است که به درستی دسته‌بندی شده است. در این جا این مقدار برابر 90.2 درصد است. در ماتریس زیر مقدار درایه‌ی سمت چپ پایین نشان دهنده‌ی تعداد داده‌ی آبی است که ما به اشتباه قرمز پیش‌بینی کرده‌ایم و مقدار درایه‌ی سمت راست بالا نشان

[3] http://www.workwithcolor.com/blue-color-hue-range-01.htm

[4] http://www.workwithcolor.com/red-color-hue-range-01.htm

دهنده‌ی تعداد داده‌ی قرمز است که ما به اشتباه آبی پیش‌بینی کرده‌ایم. اعداد روی قطر اصلی نشان دهنده‌ی تعداد نمونه‌هایی است که به درستی پیش‌بینی شده است. سمت چپ بالا تعداد نمونه قرمز و سمت راست پایین تعداد داده‌ی آبی است که درست پیش‌بینی شده است. 90.2 درصد داده‌ها به درستی دسته‌بندی شده است و ۱۲ داده به اشتباه دسته بندی شده اند که آمار آن در ماتریس آمده است. با توجه به ریز اطلاعات اشتباهات مختلف می‌توان برای بهبود دسته‌بندی اقدام کرد. در این جا برای بهبود تشخیص قرمز می‌توان اقدام کرد.



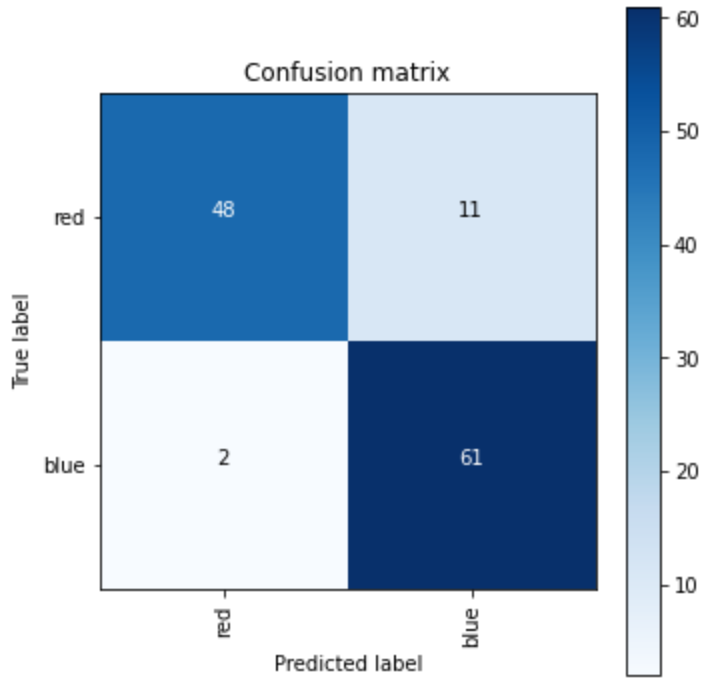

```
              precision    recall  f1-score   support

           0       0.96      0.83      0.89        58
           1       0.86      0.97      0.91        64

    accuracy                           0.90       122
   macro avg       0.91      0.90      0.90       122
weighted avg       0.91      0.90      0.90       122


The accuracy of the model is: 90.2%
```

در داده‌ها تعدادی عکس وجود داشت که لیبل آن‌ها از روی نام فایل مشخص نیست. با توجه به تصمیم طراح تمرین اگر به تمام آن‌ها لیبل قرمز دهیم نتایج زیر را خواهیم داشت.

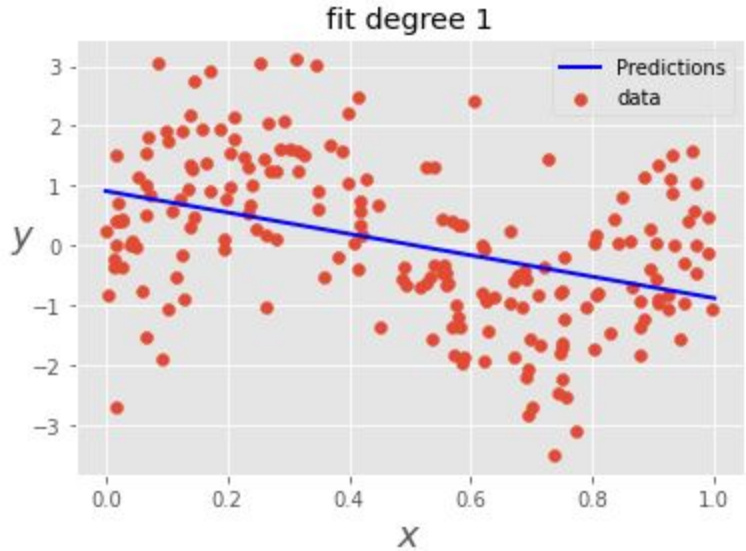


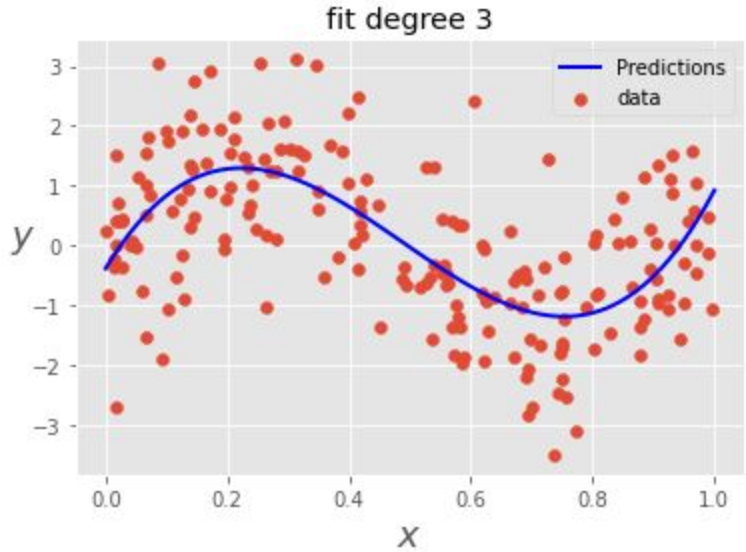
	precision	recall	f1-score	support
0	0.96	0.81	0.88	59
1	0.85	0.97	0.90	63
accuracy			0.89	122
macro avg	0.90	0.89	0.89	122
weighted avg	0.90	0.89	0.89	122

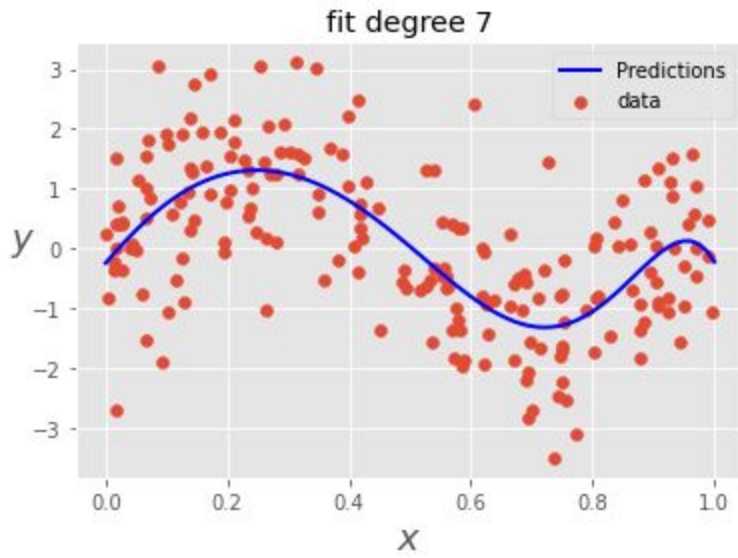
The accuracy of the model is: 89.3%

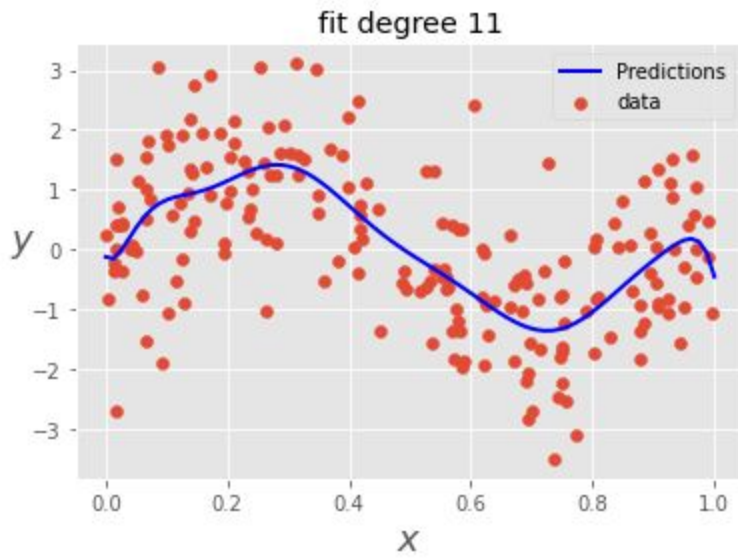
۷) ابتدا ۲۰۰ مقدار به صورت رندوم روی x بین ۰ و ۱ برای x نقاط داده انتخاب می‌کنم. سپس به ازای هر نقطه، مقدار y را از طریق $np.sin(2 * np.pi * x)$ بدست می‌آورم. برای اضافه کردن نویز، از توزیع نرمال به مرکز y بدست‌آمده و واریانس 0.16 به صورت رندم نمونه برمی‌دارم تا سیگنال سینوسی با نویز رندم نرمال مورد نظر بدست

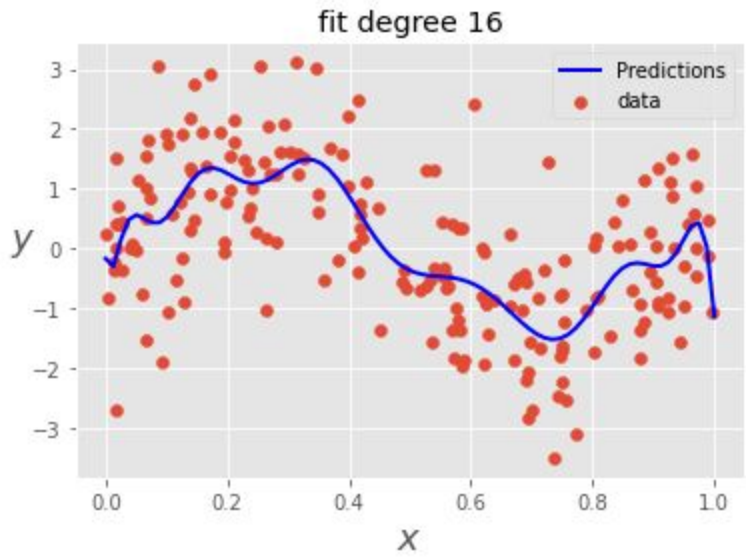
بیاید. این داده‌ها داده‌ی آموزش هستند. برای تست، تعداد ۸۰ مقدار پشت هم بین ۰ و ۱ روی x برای نقاط انتخاب می‌کنیم و y نقاط مانند قبل بدست می‌آید. به ازای هر یک از درجه‌های گفته شده، از طریق داده‌ی آموزش ضریب‌های چند جمله‌ای را بدست می‌اوریم. سپس چند جمله‌ای برازش شده و بدست آمده را برای x داده‌های تست رسم می‌کنیم. در کنار رسم چند جمله‌ای نقاط داده‌ی آموزش را نیز رسم می‌کنیم تا کیفیت برازش بر داده‌ها را بتوانیم بررسی کنیم. از y نقاط تست برای بدست‌اوردن مقدارهای خواسته شده مانند MSE استفاده می‌کنیم. نمودار برازش شده بر داده‌ها برای درجه‌های مختلف در زیر آمده است.

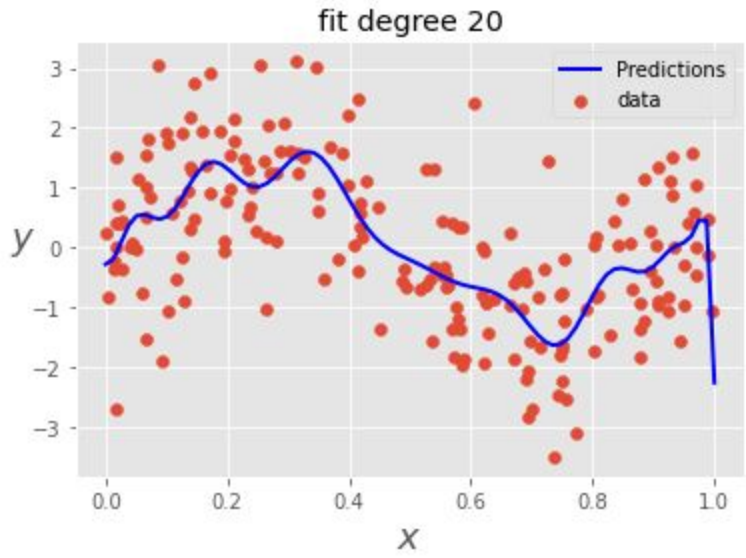





fit degree 7
Predictions
data
y
x
3
2
1
0
−1
−2
−3
0.0
0.2
0.4
0.6
0.8
1.0


fit degree 11
Predictions
data
y
x
3
2
1
0
−1
−2
−3
0.0
0.2
0.4
0.6
0.8
1.0





مقادیر خواسته شده درباره‌ی MSE، بایاس و واریانس در جدول زیر آمده است.

	MSE	var	bias^2	bias
deg = 1	1.426971	0.013796	1.413175	1.188770
deg = 3	1.260084	0.020079	1.240005	1.113555
deg = 7	1.325258	0.040535	1.284723	1.133456
deg = 11	1.392495	0.127841	1.264653	1.124568
deg = 16	2.459726	1.195218	1.264508	1.124504
deg = 20	14.060556	11.938965	2.121591	1.456568

درجه چند جمله‌ای پارامتری است که در این جا مورد بررسی قرار دادیم و آن را از مقادیر کم به زیاد افزایش دادیم. در ابتدا که درجه پایین است مثلا درجه یک، مقدار MSE بالا است که مقدار زیاد آن به دلیل بایاس بالا است. درجه‌ی یک چند جمله‌ای نمی‌تواند پیچیدگی پترن داده‌ها را برآورده کند به همین دلیل ما دچار بایاس می‌شویم که underfitting رخ می‌دهد. با بالا رفتن درجه‌ی چند جمله‌ای، به طور کلی از تاثیر بایاس در MSE کاسته می‌شود چون درجات بالاتر پیچیدگی پترن سینوسی را می‌توانند برآورده کنند. در همین شرایط با بالا رفتن درجه میزان واریانس زیاد می‌شود و در درجه‌های بالا مثل ۲۰ بیشترین تاثیر در MSE را دارد. در درجات بالا ما به برازش کردن بر روی نویز می‌پردازیم که این موضوع باعث افزایش واریانس و overfitting می‌شود. در درجه‌ی ۱۶ و ۱۱ نیز برازش بر نویز و overfitting دیده می‌شود. در درجه ۳ برازش به خوبی صورت گرفته است و در درجه ۷ کمی به برازش نویز پرداخته‌ایم. پس دیدیم که درجه چند جمله‌ای باید با توجه به پیچیدگی پترن داده‌ها انتخاب شود و نه زیاد پایین و نه زیاد بالا باشد. در درجه ۳ که برازش خوب بوده است میزان MSE در بین درجه‌های موردنظر کمینه است.

۸) به ازای هر دسته TP تعداد اشیاء متعلق به آن دسته است که به درستی دسته‌بندی شده‌اند. به ازای هر دسته FN تعداد اشیاء متعلق به آن دسته است که به اشتباه دسته‌بندی شده‌اند. به ازای هر دسته FP تعداد اشیاء است که به دسته نسبت داده شده‌اند که در حقیقت به آن تعلق ندارند.

با توجه به این تعاریف معیار Precision برای یک دسته برابر نسبت تعداد اشیاء حقیقی عضو دسته به تعداد اشیاء است که در پیش‌بینی به آن دسته نسبت داده شده است.

$$\text{Precision} = \frac{tp}{tp + fp}$$

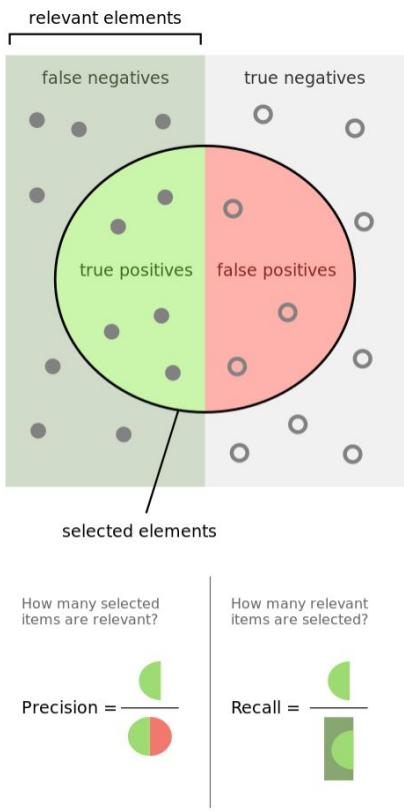
این معیار برای زمانی که تعداد FP ارزش بالایی داشته باشد یک معیار ارزشمند است. یک مثال برای این موضوع تشخیص ایمیل هرز[5] است. زمانی که تعداد نمونه‌هایی که به اشتباه به یک دسته نسبت داده شده است زیاد باشد این مقدار کم می‌شود. در این زمینه FP به معنای تشخیص یک ایمیل سالم به عنوان هرز است که تعداد زیادی از این نوع پیام‌ها باعث نارضایتی کاربر می‌شود. معیار Recall برای یک دسته برابر نسبت تعداد اشیاء حقیقی عضو دسته به تعداد اشیاء است که در واقعیت به آن دسته نسبت داده شده است.

$$\text{Recall} = \frac{tp}{tp + fn}$$

این معیار برای زمانی که تعداد FN ارزش بالایی داشته باشد یک معیار ارزشمند است. یک مثال برای این موضوع تشخیص بیماری است. FN در این شرایط به معنی عدم تشخیص بیماری فرد بیمار است که این موضوع می‌تواند باعث شیوع بیماری و به خطر افتادن جان انسان‌ها شود. معیار آخر Accuracy است که بیانگر نسبت تعداد پیش‌بینی درست به کل تعداد داده‌ها است. در هر کاری بسته به ویژگی‌های آن ممکن است یکی از recall و precision برای ارزیابی اهمیت بیشتری داشته باشد. در بیشتر مسائل معیار Accuracy به تنهایی نمی‌تواند معیار مناسبی برای عملکرد باشد. در مسئله دسته بندی رنگی اشیاء به سبز و قرمز، در عکس میعارها آمده است.

$$\text{Accuracy} = \frac{tp + tn}{tp + tn + fp + fn}$$

[5] Email spam detection


relevant elements
false negatives
true negatives
true positives
false positives
selected elements
How many selected items are relevant?
How many relevant items are selected?
Precision =
Recall =