Quiz 02

March 19, 2022

1 QUIZ 2 - Linear Regression

[ ]:
```
set.seed(1)
```

(a) Using the `rnorm()` function, create a vector, `x`, containing `100` observations drawn from a `N(0,1)` distribution. This represents a feature, `X`

[1]:
```
# x = ?

# your code here
x = rnorm(100, mean = 0, sd =1)
x
```

[2]:
```
#hidden test case
```

(b)Using the `rnorm()` function, create a vector, `eps`, containing `100` observations drawn from a `N(0,0.25)` distribution—a normal distribution with mean zero and variance `0.25`.

[3]:
```
# eps = ?

# your code here
eps = rnorm(100, mean=0, sd=0.25)
eps
```

[4]:
```
round(sd(eps),1)
```

0.2

[5]:
```
#hidden test case
```

(c)Using `x` and `eps`, generate a vector `y` according to the model

Y =−1+0.5X+ .

[6]:
```
# y = ?

# your code here
y = 1 + 0.5 * x + eps
y
```

1. 0.58469306620874 2. 0.866910241565312 3. 0.198370186372574 4. 0.629944771230657
5. 2.36879762935695 6. 0.713314219130682 7. 1.42460103061275 8. 0.894227682570246
9. 0.425517947347424 10. 1.19188104437078 11. -0.125130220854335 12. 2.21119178539693
13. 0.908001739160803 14. 1.11210255654323 15. 1.08794445749641 16. 0.785812522637756
17. 0.309572410114324 18. 1.9184836355372 19. 0.796332534862387 20. 0.725688109018953
21. 0.970731692874492 22. 1.08147452010307 23. 1.05013675632712 24. 0.130002230179903
25. 0.935256513443973 26. 1.28008054148993 27. 1.69105668993591 28. 1.43494802743776
29. 1.62591830631627 30. 0.442856852769034 31. 1.42536854432072 32. 1.20961579451079
33. 1.70962260341395 34. 0.65221389872532 35. 0.315505010803279 36. 1.64285683055199
37. 1.42710653952977 38. 0.741716406420303 39. 0.724000944778611 40. 1.7621618791357
41. 0.551445220440003 42. 1.59322057902403 43. 0.614997560657475 44. 0.211135873996296
45. 0.445207966352546 46. 0.972206820016486 47. 0.831349463764688 48. 0.18098165190928
49. 0.558472921030968 50. 0.50350728235579 51. 1.62854475681945 52. 1.16852468090341
53. 0.028601134570721 54. -0.232625725539269 55. -0.210051854723426 56. 0.884302455957182
57. 1.80898189710919 58. 0.332480738248024 59. 2.09515461444026 60. 1.03032205159314
61. 0.308132462926275 62. 1.33030454094822 63. 1.41463940456723 64. 1.59797679364242
65. 0.704928996971758 66. 1.367363114132 67. 1.41541319313962 68. 0.951378172305128
69. 0.689643403641309 70. 0.886193587634128 71. 0.778309016299188 72. 1.1623024444428
73. 0.838105312801043 74. 1.11640854120759 75. 2.23329857579195 76. 1.21924517432627
77. 1.46478126007365 78. 0.320394421349096 79. 1.6514727027383 80. 0.53210231755067
81. 2.69223579587429 82. 0.918532406160591 83. 1.05060214189714 84. 0.982324097222234
85. 0.721286670278396 86. 0.564984708312339 87. 0.10021436423514 88. 0.396744445988498
89. 1.55297752843679 90. 0.308601094185739 91. 0.984471010584043 92. 0.580953827304583
93. 0.723131661523915 94. 1.18839547123123 95. 0.0849156220087116 96. 0.941176212682495
97. 1.13537327440917 98. 0.971429823857744 99. 1.34443060183448 100. 1.87156143313845

[ ]:
```
#hidden tests
```

(d) Fit a least squares linear model to predict `y` using `x`

[9]:
```
#fit = ?

# your code here
fit = lm(y ~ x)
fit
```

```
Call:
lm(formula = y ~ x)

Coefficients:
(Intercept)            x
     0.9957       0.4941
```

[10]:
```
#hidden tests
```

(e) Now fit a polynomial regression model that predicts `y` using `x` and `x2`.

[11]:
```
#qfit = ?

# your code here
qfit = lm(y ~ x + poly(x,2))
```

[12]:
```
#hidden tests
```

[ ]:
```

```