

The screenshot shows the RStudio interface. On the left, the script pane displays the R code for `chap3.R`. The code imports a CSV file of annual sales data, performs descriptive statistics, and fits a linear regression model. The right side shows the workspace pane with variables `sales` and `results`, and the help browser pane showing the documentation for the `lm()` function.

```

95 #import a csv file of the total annual sales for each customer
96 sales <- read.csv("c:/data/yearly_sales.csv")
97
98 #examine the imported dataset
99 head(sales)
100 nrow(sales)                                #descriptive statistics
101 summary(sales)
102 #plot num_of_orders vs. sales
103 plot(sales$num_of_orders,sales$sales_total, main="Number of Orders vs. Sales")
104
105 #Perform a statistical analysis (fit a linear regression model)
106 lm <- lm(Sales$Sales_total~Sales$num_of_orders)
107 results
108
109 summary(results)
110
111 #perform some diagnostics on the fitted model
112 #plot histogram of the residuals
113 hist(results$residuals, breaks = 800)
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
709
710
711
712
713
714
715
716
717
718
719
719
720
721
722
723
724
725
726
727
728
729
729
730
731
732
733
734
735
736
737
738
739
739
740
741
742
743
744
745
746
747
748
749
749
750
751
752
753
754
755
756
757
758
759
759
760
761
762
763
764
765
766
767
768
769
769
770
771
772
773
774
775
776
777
778
779
779
780
781
782
783
784
785
786
787
787
788
789
789
790
791
792
793
794
795
796
797
797
798
799
799
800
801
802
803
804
805
806
807
808
809
809
810
811
812
813
814
815
816
817
817
818
819
819
820
821
822
823
824
825
826
827
827
828
829
829
830
831
832
833
834
835
836
837
837
838
839
839
840
841
842
843
844
845
846
846
847
848
848
849
849
850
851
852
853
854
855
856
856
857
858
858
859
859
860
861
862
863
864
865
866
866
867
868
868
869
869
870
871
872
873
874
875
876
876
877
878
878
879
879
880
881
882
883
884
885
886
886
887
888
888
889
889
890
891
892
893
894
895
895
896
896
897
897
898
898
899
899
900
901
902
903
904
905
906
907
907
908
909
909
910
911
912
913
913
914
914
915
915
916
916
917
917
918
918
919
919
920
920
921
921
922
922
923
923
924
924
925
925
926
926
927
927
928
928
929
929
930
930
931
931
932
932
933
933
934
934
935
935
936
936
937
937
938
938
939
939
940
940
941
941
942
942
943
943
944
944
945
945
946
946
947
947
948
948
949
949
950
950
951
951
952
952
953
953
954
954
955
955
956
956
957
957
958
958
959
959
960
960
961
961
962
962
963
963
964
964
965
965
966
966
967
967
968
968
969
969
970
970
971
971
972
972
973
973
974
974
975
975
976
976
977
977
978
978
979
979
980
980
981
981
982
982
983
983
984
984
985
985
986
986
987
987
988
988
989
989
990
990
991
991
992
992
993
993
994
994
995
995
996
996
997
997
998
998
999
999
1000
1000
1001
1001
1002
1002
1003
1003
1004
1004
1005
1005
1006
1006
1007
1007
1008
1008
1009
1009
1010
1010
1011
1011
1012
1012
1013
1013
1014
1014
1015
1015
1016
1016
1017
1017
1018
1018
1019
1019
1020
1020
1021
1021
1022
1022
1023
1023
1024
1024
1025
1025
1026
1026
1027
1027
1028
1028
1029
1029
1030
1030
1031
1031
1032
1032
1033
1033
1034
1034
1035
1035
1036
1036
1037
1037
1038
1038
1039
1039
1040
1040
1041
1041
1042
1042
1043
1043
1044
1044
1045
1045
1046
1046
1047
1047
1048
1048
1049
1049
1050
1050
1051
1051
1052
1052
1053
1053
1054
1054
1055
1055
1056
1056
1057
1057
1058
1058
1059
1059
1060
1060
1061
1061
1062
1062
1063
1063
1064
1064
1065
1065
1066
1066
1067
1067
1068
1068
1069
1069
1070
1070
1071
1071
1072
1072
1073
1073
1074
1074
1075
1075
1076
1076
1077
1077
1078
1078
1079
1079
1080
1080
1081
1081
1082
1082
1083
1083
1084
1084
1085
1085
1086
1086
1087
1087
1088
1088
1089
1089
1090
1090
1091
1091
1092
1092
1093
1093
1094
1094
1095
1095
1096
1096
1097
1097
1098
1098
1099
1099
1100
1100
1101
1101
1102
1102
1103
1103
1104
1104
1105
1105
1106
1106
1107
1107
1108
1108
1109
1109
1110
1110
1111
1111
1112
1112
1113
1113
1114
1114
1115
1115
1116
1116
1117
1117
1118
1118
1119
1119
1120
1120
1121
1121
1122
1122
1123
1123
1124
1124
1125
1125
1126
1126
1127
1127
1128
1128
1129
1129
1130
1130
1131
1131
1132
1132
1133
1133
1134
1134
1135
1135
1136
1136
1137
1137
1138
1138
1139
1139
1140
1140
1141
1141
1142
1142
1143
1143
1144
1144
1145
1145
1146
1146
1147
1147
1148
1148
1149
1149
1150
1150
1151
1151
1152
1152
1153
1153
1154
1154
1155
1155
1156
1156
1157
1157
1158
1158
1159
1159
1160
1160
1161
1161
1162
1162
1163
1163
1164
1164
1165
1165
1166
1166
1167
1167
1168
1168
1169
1169
1170
1170
1171
1171
1172
1172
1173
1173
1174
1174
1175
1175
1176
1176
1177
1177
1178
1178
1179
1179
1180
1180
1181
1181
1182
1182
1183
1183
1184
1184
1185
1185
1186
1186
1187
1187
1188
1188
1189
1189
1190
1190
1191
1191
1192
1192
1193
1193
1194
1194
1195
1195
1196
1196
1197
1197
1198
1198
1199
1199
1200
1200
1201
1201
1202
1202
1203
1203
1204
1204
1205
1205
1206
1206
1207
1207
1208
1208
1209
1209
1210
1210
1211
1211
1212
1212
1213
1213
1214
1214
1215
1215
1216
1216
1217
1217
1218
1218
1219
1219
1220
1220
1221
1221
1222
1222
1223
1223
1224
1224
1225
1225
1226
1226
1227
1227
1228
1228
1229
1229
1230
1230
1231
1231
1232
1232
1233
1233
1234
1234
1235
1235
1236
1236
1237
1237
1238
1238
1239
1239
1240
1240
1241
1241
1242
1242
1243
1243
1244
1244
1245
1245
1246
1246
1247
1247
1248
1248
1249
1249
1250
1250
1251
1251
1252
1252
1253
1253
1254
1254
1255
1255
1256
1256
1257
1257
1258
1258
1259
1259
1260
1260
1261
1261
1262
1262
1263
1263
1264
1264
1265
1265
1266
1266
1267
1267
1268
1268
1269
1269
1270
1270
1271
1271
1272
1272
1273
1273
1274
1274
1275
1275
1276
1276
1277
1277
1278
1278
1279
1279
1280
1280
1281
1281
1282
1282
1283
1283
1284
1284
1285
1285
1286
1286
1287
1287
1288
1288
1289
1289
1290
1290
1291
1291
1292
1292
1293
1293
1294
1294
1295
1295
1296
1296
1297
1297
1298
1298
1299
1299
1300
1300
1301
1301
1302
1302
1303
1303
1304
1304
1305
1305
1306
1306
1307
1307
1308
1308
1309
1309
1310
1310
1311
1311
1312
1312
1313
1313
1314
1314
1315
1315
1316
1316
1317
1317
1318
1318
1319
1319
1320
1320
1321
1321
1322
1322
1323
1323
1324
1324
1325
1325
1326
1326
1327
1327
1328
1328
1329
1329
1330
1330
1331
1331
1332
1332
1333
1333
1334
1334
1335
1335
1336
1336
1337
1337
1338
1338
1339
1339
1340
1340
1341
1341
1342
1342
1343
1343
1344
1344
1345
1345
1346
1346
1347
1347
1348
1348
1349
1349
1350
1350
1351
1351
1352
1352
1353
1353
1354
1354
1355
1355
1356
1356
1357
1357
1358
1358
1359
1359
1360
1360
1361
1361
1362
1362
1363
1363
1364
1364
1365
1365
1366
1366
1367
1367
1368
1368
1369
1369
1370
1370
1371
1371
1372
1372
1373
1373
1374
1374
1375
1375
1376
1376
1377
1377
1378
1378
1379
1379
1380
1380
1381
1381
1382
1382
1383
1383
1384
1384
1385
1385
1386
1386
1387
1387
1388
1388
1389
1389
1390
1390
1391
1391
1392
1392
1393
1393
1394
1394
1395
1395
1396
1396
1397
1397
1398
1398
1399
1399
1400
1400
1401
1401
1402
1402
1403
1403
1404
1404
1405
1405
1406
1406
1407
1407
1408
1408
1409
1409
1410
1410
1411
1411
1412
1412
1413
1413
1414
1414
1415
1415
1416
1416
1417
1417
1418
1418
1419
1419
1420
1420
1421
1421
1422
1422
1423
1423
1424
1424
1425
1425
1426
1426
1427
1427
1428
1428
1429
1429
1430
1430
1431
1431
1432
1432
1433
1433
1434
1434
1435
1435
1436
1436
1437
1437
1438
1438
1439
1439
1440
1440
1441
1441
1442
1442
1443
1443
1444
1444
1445
1445
1446
1446
1447
1447
1448
1448
1449
1449
1450
1450
1451
1451
1452
1452
1453
1453
1454
1454
1455
1455
1456
1456
1457
1457
1458
1458
1459
1459
1460
1460
1461
1461
1462
1462
1463
1463
1464
1464
1465
1465
1466
1466
1467
1467
1468
1468
1469
1469
1470
1470
1471
1471
1472
1472
1473
1473
1474
1474
1475
1475
1476
1476
1477
1477
1478
1478
1479
1479
1480
1480
1481
1481
1482
1482
1483
1483
1484
1484
1485
1485
1486
1486
1487
1487
1488
1488
1489
1489
1490
1490
1491
1491
1492
1492
1493
1493
1494
1494
1495
1495
1496
1496
1497
1497
1498
1498
1499
1499
1500
1500
1501
1501
1502
1502
1503
1503
1504
1504
1505
1505
1506
1506
1507
1507
1508
1508
1509
1509
1510
1510
1511
1511
1512
1512
1513
1513
1514
1514
1515
1515
1516
1516
1517
1517
1518
1518
1519
1519
1520
1520
1521
1521
1522
1522
1523
1523
1524
1524
1525
1525
1526
1526
1527
1527
1528
1528
1529
1529
1530
1530
1531
1531
1532
1532
1533
1533
1534
1534
1535
1535
1536
1536
1537
1537
1538
1538
1539
1539
1540
1540
1541
1541
1542
1542
1543
1543
1544
1544
1545
1545
1546
1546
1547
1547
1548
1548
1549
1549
1550
1550
1551
1551
1552
1552
1553
1553
1554
1554
1555
1555
1556
1556
1557
1557
1558
1558
1559
1559
1560
1560
1561
1561
1562
1562
1563
1563
1564
1564
1565
1565
1566
1566
1567
1567
1568
1568
1569
1569
1570
1570
1571
1571
1572
1572
1573
1573
1574
1574
1575
1575
1576
1576
1577
1577
1578
1578
1579
1579
1580
1580
1581
1581
1582
1582
1583
1583
1584
1584
1585
1585
1586
1586
1587
1587
1588
1588
1589
1589
1590
1590
1591
1591
1592
1592
1593
1593
1594
1594
1595
1595
1596
1596
1597
1597
1598
1598
1599
1599
1600
1600
1601
1601
1602
1602
1603
1603
1604
1604
1605
1605
1606
1606
1607
1607
1608
1608
1609
1609
1610
1610
1611
1611
1612
1612
1613
1613
1614
1614
1615
1615
1616
1616
1617
1617
1618
1618
1619
1619
1620
1620
1621
1621
1622
1622
1623
1623
1624
1624
1625
1625
1626
1626
1627
1627
1628
1628
1629
1629
1630
1630
1631
1631
1632
1632
1633
1633
1634
1634
1635
1635
1636
1636
1637
1637
1638
1638
1639
1639
1640
1640
1641
1641
1642
1642
1643
1643
1644
1644
1645
1645
1646
1646
1647
1647
1648
1648
1649
1649
1650
1650
1651
1651
1652
1652
1653
1653
1654
1654
1655
1655
1656
1656
1657
1657
1658
1658
1659
1659
1660
1660
1661
1661
1662
1662
1663
1663
1664
1664
1665
1665
1666
1666
1667
1667
1668
1668
1669
1669
1670
1670
1671
1671
1672
1672
1673
1673
1674
1674
1675
1675
1676
1676
1677
1677
1678
1678
1679
1679
1680
1680
1681
1681
1682
1682
1683
1683
1684
1684
1685
1685
1686
1686
1687
1687
1688
1688
1689
1689
1690
1690
1691
1691
1692
1692
1693
1693
1694
1694
1695
1695
1696
1696
1697
1697
1698
1698
1699
1699
1700
1700
1701
1701
1702
1702
1703
1703
1704
1704
1705
1705
1706
1706
1707
1707
1708
1708
1709
1709
1710
1710
1711
1711
1712
1712
1713
1713
1714
1714
1715
1715
1716
1716
1717
1717
1718
1718
1719
1719
1720
1720
1721
1721
1722
1722
1723
1723
1724
1724
1725
1725
1726
1726
1727
1727
1728
1728
1729
1729
1730
1730
1731
1731
1732
1732
1733
1733
1734
1734
1735
1735
1736
1736
1737
1737
1738
1738
1739
1739
1740
1740
1741
1741
1742
1742
1743
1743
1744
1744
1745
1745
1746
1746
1747
1747
1748
1748
1749
1749
1750
1750
1751
1751
1752
1752
1753
1753
1754
1754
1755
1755
1756
1756
1757
1757
1758
1758
1759
1759
1760
1760
1761
1761
1762
1762
1763
1763
1764
1764
1765
1765
1766
1766
1767
1767
1768
1768
1769
1769
1770
1770
1771
1771
1772
1772
1773
1773
1774
1774
1775
1775
1776
1776
1777
1777
1778
1778
1779
1779
1780
1780
1781
1781
1782
1782
1783
1783
1784
1784
1785
1785
1786
1786
1787
1787
1788
1788
1789
1789
1790
1790
1791
1791
1792
1792
1793
1793
1794
1794
1795
1795
1796
1796
179
```

en un archivo de datos utiliza una coma para el decimal, R también proporciona dos funciones adicionales: `read.csv2()` y `read.delim2()` —Para importar dichos datos. La tabla 3-1 incluye los valores predeterminados esperados para los encabezados, los separadores de columnas y las notaciones de punto decimal.

T Poder 3-1 Importar valores predeterminados de funciones

<code>read.table()</code>	FALSO	""	""
<code>read.csv()</code>	CIERTO	","	"."
<code>read.csv2()</code>	CIERTO	","	","
<code>read.delim()</code>	CIERTO	"\ T"	"."
<code>read.delim2()</code>	CIERTO	"\ T"	","

Las funciones R análogas como `write.table()`, `write.csv()`, y `write.csv2()` habilitan la exportación de conjuntos de datos de R a un archivo externo. Por ejemplo, el siguiente código R agrega una columna adicional al conjunto de datos de ventas y exporta el conjunto de datos modificado a un archivo externo.

```
# agregue una columna para las ventas promedio por pedido
ventas $ per_order <- ventas $ sales_total / sales $ num_of_orders
```

```
# exportar datos como delimitados por tabuladores sin los nombres de las filas
write.table(ventas, "sales_modified.txt", sep = "\t", row.names = FALSE)
```

A veces es necesario leer datos de un sistema de gestión de bases de datos (DBMS). Paquetes R como DBI [6] y RODBC [7] están disponibles para este propósito. Estos paquetes proporcionan interfaces de base de datos para la comunicación entre R y DBMS como MySQL, Oracle, SQL Server, PostgreSQL y Pivotal Greenplum. El siguiente código R demuestra cómo instalar el RODBC paquete con el Instalar en pc

. paquetes () función. los biblioteca() carga el paquete en el espacio de trabajo de R. Finalmente, un conector (**conexión**) se inicializa para conectarse a una base de datos de Pivotal Greenplum **entrenamiento2** a través de la conectividad de base de datos abierta (ODBC) con el usuario usuario. los **entrenamiento2** base de datos debe definirse en el

/etc/ODBC.ini archivo de configuración o utilizando las Herramientas administrativas del Panel de control de Windows.

```
install.packages ("RODBC")
biblioteca (RODBC)
conn <- odbcConnect ("entrenamiento2", uid = "usuario", pwd = "contraseña")
```

El conector debe estar presente para enviar una consulta SQL a una base de datos ODBC utilizando el `sqlQuery()` función de la RODBC paquete. El siguiente código R recupera columnas específicas del **alojamiento** tabla en la que los ingresos del hogar (`hinc`) es superior a \$ 1,000,000.

```
vivienda_datos <- sqlQuery (conn, "seleccionar serialno, estado, personas, habitaciones
de la vivienda
donde hinc > 1000000 ")
head (datos_de_casa)
serialno estado personas habitaciones 1
3417867 6 2 7
2 3417867 6 2 7
```

3 4552088	6	5	9
4 4552088	6	5	9
5 8699293	6	5	5
6 8699293	6	5	5

Aunque los gráficos se pueden guardar usando la GUI de RStudio, los gráficos también se pueden guardar usando el código R especificando los dispositivos gráficos apropiados. Utilizando la `jpeg()`, el siguiente código R crea un nuevo archivo JPEG, agrega un `histogramplot` al archivo y luego cierra el archivo. Estas técnicas son útiles al automatizar informes estándar. Otras funciones, como `png()`, `bmp()`, `pdf()`, y `psddata()`, están disponibles en R para guardar gráficos en el formato deseado.

```
jpeg (archivo = "c:/data/sales_hist.jpeg") # crear un nuevo archivo jpeg
hist (ventas $ num_of_orders) # exportar histograma a jpeg
dev.off () # apague el dispositivo gráfico
```

Puede encontrar más información sobre importaciones y exportaciones de datos en <http://cran.r-project.org/doc/manuals/r-release/R-data.html>, por ejemplo, cómo importar conjuntos de datos de paquetes de software estadístico, incluidos Minitab, SAS y SPSS.

3.1.3 Atributos y tipos de datos

En el ejemplo anterior, el `ventas` La variable contenía un registro para cada cliente. Se proporcionaron varias características para cada cliente, como las ventas anuales totales, el número de pedidos y el género. En general, estas características o atributos proporcionan las medidas cualitativas y cuantitativas para cada ítem o tema de interés. Los atributos se pueden clasificar en cuatro tipos: nominal, ordinal, intervalo y razón (NOIR) [8]. La Tabla 3-2 distingue estos cuatro tipos de atributos y muestra las operaciones que soportan. Los atributos nominales y ordinales se consideran atributos categóricos, mientras que los atributos de intervalo y razón se consideran atributos numéricos.

T PODER 3-2 *Tipos de atributos NOIR*

	Nominal	Ordinal	Intervalo	Proporción
Definición	Los valores representan etiquetas que distinguen <u>Guish uno de otro.</u>	Atributos implicar una secuencia.	La diferencia entre dos los valores son significativo.	Tanto la diferencia y la proporción de dos valores son significativo.
Ejemplos	Códigos postales, nacional idy, nombres de las calles, género, ID de empleado números, CIERTO o FALSO	Calidad de diamantes académico grados, magnitud de temblores	Temperatura en Celsius o Fahrenheit, calendario fechas, latitudes	Edad, temperatura en Kelvin, cuenta, longitud, peso
Operaciones	=, ≠	=, ≠, <, ≤, >, ≥	=, ≠, <, ≤, >, ≥, +, -	=, ≠, <, ≤, >, ≥, +, - , ×, ÷

Los datos de un tipo de atributo pueden convertirse en otro. Por ejemplo, el *calidad* de diamantes {Regular, Bueno, Muy Bueno, Premium, Ideal} se considera ordinal pero se puede convertir a {Bueno, Excelente} nominal con un mapeo de fi nido. Del mismo modo, un atributo de relación como *Años* se puede convertir en un atributo ordinal como {Infant, Adolescent, Adult, Senior}. Comprender los tipos de atributos en un conjunto de datos determinado es importante para garantizar que se apliquen e interpreten correctamente las estadísticas descriptivas y los métodos analíticos adecuados. Por ejemplo, la desviación estándar y estándar de los códigos postales de EE. UU. No es muy significativa ni apropiada. El manejo adecuado de las variables categóricas se abordará en los capítulos siguientes. Además, es útil considerar estos tipos de atributos durante la siguiente discusión sobre los tipos de datos R.

Tipos de datos numéricos, de caracteres y lógicos

Al igual que otros lenguajes de programación, R admite el uso de valores numéricos, de caracteres y lógicos (booleanos). En el siguiente código R se dan ejemplos de tales variables.

```
yo <- 1                      # crea una variable numérica
deporte <- "fútbol"           # crea una variable de carácter
bandera <- VERDADERO         # crea una variable lógica
```

R proporciona varias funciones, como *clase()* y *tipo de()*, para examinar las características de una variable dada. La función *clase()* representa la clase abstracta de un objeto. La función *tipo de()* determina la forma en que un objeto se almacena en la memoria. A pesar de que *yo* parece ser un número entero, *yo* se almacena internamente con doble precisión. Para mejorar la legibilidad de los segmentos de código en esta sección, los comentarios de R en línea se utilizan para explicar el código o para proporcionar los valores devueltos.

```
clase (i)                      # devuelve "numérico"
tipo de (i)                     # devuelve "doble"

clase (deporte)                # devuelve "carácter"
typeof (deporte)               # devuelve "carácter"

clase (bandera)                # devuelve "lógico"
typeof (bandera)               # devuelve "lógico"
```

Existen funciones R adicionales que pueden probar las variables y convertir una variable en un tipo específico. El siguiente código R ilustra cómo probar si *yo* es un número entero usando el *es.integer()* función y coaccionar *yo* en una nueva variable entera, *j*, utilizando la *como entero()* función. Se pueden aplicar funciones similares para tipos dobles, de caracteres y lógicos.

```
es un entero (i)              # devuelve FALSO
j <- como entero (i)          # convierte el contenido de i en un número entero
es.integer (j)                # devuelve VERDADERO
```

La aplicación de la *longitud()* función revela que las variables creadas tienen cada una una longitud de 1. Se podría haber esperado la longitud devuelta de *deporte* haber sido 8 para cada uno de los caracteres de la cadena "fútbol americano". Sin embargo, estas tres variables son en realidad un elemento, *vectores*.

```
longitud (i)                  # devuelve 1
longitud (bandera)            # devuelve 1
longitud (deporte)            # devuelve 1 (no 8 para "fútbol")
```

Vectores

Los vectores son un bloque de construcción básico para los datos en R. Como se vio anteriormente, las variables R simples son en realidad vectores.

Un vector solo puede constar de valores de la misma clase. Las pruebas de vectores se pueden realizar utilizando el `es.vector()` función.

```
es.vector (i)                                # devuelve VERDADERO
es.vector (bandera)                          # devuelve VERDADERO
is.vector (deporte)                           # devuelve VERDADERO
```

R proporciona una funcionalidad que permite la fácil creación y manipulación de vectores. El siguiente código R ilustra cómo se puede crear un vector usando la función de combinación, `C()` o el operador de dos puntos`:`, para construir un vector a partir de la secuencia de números enteros del 1 al 5. Además, el código muestra cómo se pueden modificar o acceder fácilmente a los valores de un vector existente. El código, relacionado con el `z` vector, indica cómo se pueden construir comparaciones lógicas para extraer ciertos elementos de un vector dado.

```
u <- c ("rojo", "amarillo", "azul") # crear un vector "rojo" "amarillo" "azul"
tu                                     # devuelve "rojo" "amarillo" "azul"
u [1]                                  # devuelve "rojo" (primer elemento en u)
v <- 1: 5                               # crear un vector 1 2 3 4 5
v                                     # devuelve 1 2 3 4 5
suma (v)                               # devuelve 15
w <- v * 2                             # crear un vector 2 4 6 8 10
w                                     # devuelve 2 4 6 8 10
w [3]                                  # devuelve 6 (el tercer elemento de w)
z <- v + w                            # suma dos vectores elemento por elemento
z                                     # devuelve 3 6 9 12 15
z > 8                                  # devuelve FALSO FALSO VERDADERO VERDADERO VERDADERO
z [z > 8]                              # devuelve 9 12 15
z [z > 8 | z <5]                      # devuelve 3 9 12 15 ("|" denota "o")
```

A veces es necesario inicializar un vector de una longitud específica y luego poblar el contenido del vector más tarde. La función, por defecto, crea un vector lógico. Se puede especificar un vector de un tipo diferente usando el modo parámetro. El vector `C`, un vector entero de longitud 0, puede ser útil cuando el número de elementos no se conoce inicialmente y los nuevos elementos se agregarán más tarde al final del vector cuando los valores estén disponibles.

```
a <- vector (longitud = 3)                # crea un vector lógico de longitud 3
un                                     # devuelve FALSE FALSE FALSE
b <- vector (modo = "numérico", 3) tipo de (b) # crea un vector numérico de longitud 3
                                              # devuelve "doble"
b [2] <- 3.1                            # asignar 3.1 al segundo elemento
segundo                                # devuelve 0.0 3.1 0.0
c <- vector (modo = "entero", 0) c        # crea un vector entero de longitud 0
                                              # devuelve entero (0)
longitud (c)                           # devuelve 0
```

Aunque los vectores pueden parecer análogos a las matrices de una dimensión, técnicamente son adimensionales, como se ve en el siguiente código R. El concepto de matrices y matrices se aborda en la siguiente discusión.

longitud (b)	# devuelve 3
tenue (b)	# devuelve NULL (un valor indefinido)

Matrices y matrices

los formación() La función se puede utilizar para reestructurar un vector como una matriz. Por ejemplo, el siguiente código R crea una matriz tridimensional para contener las ventas trimestrales de tres regiones durante un período de dos años y luego asigna la cantidad de ventas de \$ 158,000 a la segunda región para el primer trimestre del primer año.

```
# las dimensiones son 3 regiones, 4 trimestres y 2 años
ventas_trimestrales <- array (0, dim = c (3,4,2))
ventas_trimestrales [2,1,1] <- 158000
ventas_trimestrales
```

,, 1

```
[, 1] [, 2] [, 3] [, 4]
[1,]      0      0      0      0
[2,] 158000      0      0      0
[3,]      0      0      0      0
```

,, 2

```
[, 1] [, 2] [, 3] [, 4]
[1,]      0      0      0      0
[2,]      0      0      0      0
[3,]      0      0      0      0
```

Una matriz bidimensional se conoce como **matriz**. El siguiente código inicializa una matriz para mantener las ventas trimestrales de las tres regiones. Los parámetros nrow y ncol definen el número de filas y columnas, respectivamente, para el **sales_matrix**.

```
sales_matrix <- matriz (0, nrow = 3, ncol = 4) sales_matrix
```

```
[, 1] [, 2] [, 3] [, 4]
[1,]      0      0      0      0
[2,]      0      0      0      0
[3,]      0      0      0      0
```

R proporciona las operaciones matriciales estándar, como suma, resta y multiplicación, así como la función de transposición. t () y la función de matriz inversa matrix.inverse () incluido en el matrixcalc paquete. El siguiente código de R crea un 3×3 , M, y la multiplica por su inversa para obtener la matriz identidad.

```
biblioteca (matrixcalc)
M <- matriz (c (1,3,3,5,0,4,3,3,3), nrow = 3, ncol = 3) # construir una matriz de 3x3
```

```
M% *% matriz inversa (M) # multiplica M por inverso (M)
```

```
[, 1] [, 2] [, 3]
[1,]     1     0     0
[2,]     0     1     0
[3,]     0     0     1
```

Marcos de datos

De forma similar al concepto de matrices, los marcos de datos proporcionan una estructura para almacenar y acceder a varias variables de posiblemente diferentes tipos de datos. De hecho, como `is.data.frame()` función indica, un marco de datos fue creado por el `read.csv()` función al comienzo del capítulo.

```
# importar un archivo CSV de las ventas anuales totales de cada cliente
ventas <- read.csv ("c:/data/yearly_sales.csv")
is.data.frame (ventas) # devuelve VERDADERO
```

Como se vio anteriormente, se puede acceder fácilmente a las variables almacenadas en el marco de datos usando la notación `$`. El siguiente código R ilustra que en este ejemplo, cada variable es un vector con la excepción de género, que fue, por un `read.csv()` predeterminado, importado como **factor**. Analizado en detalle más adelante en esta sección, un factor denota una variable categórica, típicamente con unos pocos niveles finitos como "F" y "M" en el caso de **género**.

```
length (ventas $ num_of_orders) # devuelve 10000 (número de clientes)
```

```
es.vector (ventas $ cust_id) # devuelve VERDADERO
es.vector (ventas $ sales_total) # devuelve VERDADERO
es.vector (ventas $ num_of_orders) # devuelve VERDADERO
es.vector (ventas $ género) # devuelve FALSO
```

```
is.factor (ventas $ género) # devuelve VERDADERO
```

Debido a su flexibilidad para manejar muchos tipos de datos, los marcos de datos son el formato de entrada preferido para muchas de las funciones de modelado disponibles en R. El siguiente uso de la `str()` función proporciona la estructura de la **ventas** marco de datos. Esta función identifica los tipos de datos enteros y numéricos (dobles), las variables de factor y los niveles, así como los primeros valores de cada variable.

```
str (ventas) # estructura de visualización del objeto de marco de datos
```

```
'data.frame': 10000 obs. de 4 variables: $ cust_id
  : int 100001 100002 100003 100004 100005 100006 ...
$ sales_total: num 800.6 217.5 74.6 498.6 723.1 ...
$ num_of_orders: int 3 3 2 3 4 2 2
  2 2 2 ...
$ gender
  : Factor con 2 niveles "F", "M": 1 1 2 2 1 1 2 2 1 2 ...
```

En el sentido más simple, los marcos de datos son listas de variables de la misma longitud. Un subconjunto del marco de datos se puede recuperar mediante **operadores de subconjuntos**. Los operadores de subconjuntos de R son poderosos porque permiten expresar operaciones complejas de manera sencilla y recuperar fácilmente un subconjunto del conjunto de datos.

```
# extrae la cuarta columna del marco de datos de ventas
ventas [, 4]
# extraer la columna de género del marco de datos de ventas
```

```
ventas $ género
# recuperar las dos primeras filas del marco de datos
ventas [1: 2]
# recuperar la primera, tercera y cuarta columnas
ventas [, c (1,3,4)]
# recupere las columnas cust_id y sales_total
ventas [, c ("cust_id", "sales_total")]
# recuperar todos los registros cuyo género sea femenino
ventas [ventas $ género == "F".]
```

El siguiente código R muestra que la clase de ventas variable es un marco de datos. Sin embargo, el tipo de ventas variable es una lista. UN *lista* es una colección de objetos que pueden ser de varios tipos, incluidas otras listas.

```
clase (ventas)
"marco de datos"
typeof (ventas)
"lista"
```

Liza

Las listas pueden contener cualquier tipo de objeto, incluidas otras listas. Usando el vector v y la matriz M creada en ejemplos anteriores, el siguiente código R crea *surtido*, una lista de diferentes tipos de objetos.

```
# construya una lista variada de una cadena, un numérico, una lista, un vector,
# y una matriz
vivienda <- lista ("propia", "alquiler")
surtido <- lista ("fútbol", 7.5, vivienda, v, M) surtido
```

```
[[1]]
[1] "fútbol"
```

```
[[2]]
[1] 7.5
```

```
[[3]]
[[3]][[1]]
[1] "propio"
```

```
[[3]][[2]]
[1] "alquilar"
```

```
[[4]]
[1] 1 2 3 4 5
```

```
[[5]]
```

```
[, 1] [, 2] [, 3]
[1,]     1     5     3
[2,]     3     0     3
[3,]     3     4     3
```

Al mostrar el contenido de **surtido**, el uso de corchetes dobles, [[]], es de particular importancia. Como ilustra el siguiente código R, el uso de un único conjunto de corchetes solo accede a un elemento de la lista, no a su contenido.

```
# examinar el quinto objeto, M, en la lista
clase (surtido [5])                                # devuelve "lista"
longitud (surtido [5])                             # devuelve 1

clase (surtido [[5]])                               # devuelve "matriz"
longitud (surtido [[5]])                           # devuelve 9 (para la matriz 3x3)
```

Como se presentó anteriormente en la discusión del marco de datos, el str () función ofrece detalles sobre la estructura de una lista.

```
str (surtido)
Lista de 5
$: chr "fútbol" $: num 7.5

$: Lista de 2
.. $.: chr "propio"
.. $.: chr "alquiler"
$: int [1: 5] 1 2 3 4 5
$: num [1: 3, 1: 3] 1 3 3 5 0 4 3 3 3
```

Factores

Los factores se introdujeron brevemente durante la discusión de la **género** variable en el marco de datos **ventas**.

En este caso, **género** podría asumir uno de dos niveles: F o METRO. Los factores pueden ordenarse o no. En el caso de **género**, los niveles no están ordenados.

```
clase (ventas $ género)                         # devuelve "factor"
is.ordered (ventas $ género)                   # devuelve FALSO
```

Incluido con el ggplot2 paquete, el **diamantes** El marco de datos contiene tres factores ordenados. Examinando el **cortar** factor, hay cinco niveles para mejorar el corte: Regular, Bueno, Muy Bueno, Premium e Ideal. Así, ventas \$ género contiene datos nominales, y diamantes \$ cortados contiene datos ordinales.

```
cabeza (ventas $ género)                      # mostrar los primeros seis valores y los niveles
```

FFMMFF

Niveles: FM

```
biblioteca (ggplot2)
datos (diamantes)                            # carga el marco de datos en el espacio de trabajo de R
```

```
str (diamantes)
```

```
'data.frame': 53940 obs. de 10 variables:
```

```
 $ quilate: num 0.23 0.21 0.23 0.29 0.31 0.24 0.24 0.26 0.22 ... $ corte
```

```
 : Factor de pedido con 5 niveles "Regular" <"Bueno" <..: 5 4 2 4 2 3 ...
```

```
 $ color: factor ordinal con 7 niveles "D" <"E" <"F" <"G" <..: 2 2 2 6 7 7 ... $ claridad: factor ordinal con 8
```

```
niveles "I1" <" SI2" <" SI1 " <..: 2 3 5 4 2 ... $ profundidad: num 61,5 59,8 56,9 62,4 63,3 62,8 62,3 61,9 65,1
```

```
59,4 ... $ tabla: num 55 61 65 58 58 57 57 55 61 61 ...
```

```
$ precio: int 326 326 327 334 335 336 336 337 337 338 ... $ x
```

```
: num 3.95 3.89 4.05 4.2 4.34 3.94 3.95 4.07 3.87 4 ... $ y
```

```
: num 3.98 3.84 4.07 4.23 4.35 3.96 3.98 4.11 3.78 4.05 $ z
```

```
: num 2.43 2.31 2.31 2.63 2.75 2.48 2.47 2.53 2.49 2.39
```

```
...  
...
```

```
cabeza (diamantes $ cortados) # mostrar los primeros seis valores y los niveles
```

```
Ideal Prima Bueno Prima Bueno Muy buena
```

Niveles: Regular <Bueno <Muy bueno <Premium <ideal

Supongamos que se decide categorizar **ventas \$ sales_totals** en tres grupos —pequeño, mediano y grande— según el monto de las ventas con el siguiente código. Estas agrupaciones son la base del nuevo factor ordinal, gastador, con niveles {pequeño, mediano, grande}.

```
# construir un vector de caracteres vacío de la misma longitud que las ventas
```

```
sales_group <- vector (mode = "personaje",
```

```
length = length (ventas $ sales_total))
```

```
# agrupar a los clientes según el monto de las ventas
```

```
sales_group [ventas $ sales_total <100] <- "pequeño"
```

```
sales_group [ventas $ sales_total >= 100 & ventas $ sales_total <500] <- "medium" sales_group [ventas $
```

```
sales_total >= 500] <- "grande"
```

```
# crear y agregar el factor ordenado al marco de datos de ventas
```

```
gastador <- factor (grupo_ventas, niveles = c ("pequeño", "mediano", "grande"),
```

```
ordenado = VERDADERO)
```

```
ventas <- cbind (ventas, gastador)
```

```
str (gastador de $ ventas)
```

Factor de pedido con 3 niveles "pequeño" <"medio" <..: 3 2 1 2 3 1 1 1 2 1 ...

```
cabeza (ventas $ gastador)
```

```
grande mediano pequeño mediano grande pequeño
```

Niveles: pequeño <mediano <grande

los cbind () La función se utiliza para combinar variables en columnas. los rbind () La función se utiliza para combinar conjuntos de datos en filas. El uso de factores es importante en varias funciones de modelado estadístico R, como el análisis de varianza, aov (), que se presenta más adelante en este capítulo, y el uso de tablas de contingencia, que se analiza a continuación.

Tablas de contingencia

En R, **mesa** se refiere a una clase de objetos utilizados para almacenar los recuentos observados en los factores para un conjunto de datos dado. Dicha tabla se conoce comúnmente como tabla de contingencia y es la base para realizar una prueba estadística sobre la independencia de los factores utilizados para construir la tabla. El siguiente código R crea una tabla de contingencia basada en el ventas \$ género y ventas \$ gastador factores.

```
# construir una tabla de contingencia basada en el género y los factores del gasto
```

```
sales_table <- table(ventas ~ género, ventas ~ gastador)
```

```
sales_table
```

```
  pequeño mediano grande
F 1726      2746 563
M 1656      2723586
```

```
clase(tabla_ventas)           # devuelve "tabla"
```

```
typeof(tabla_ventas)          # devuelve "entero"
```

```
tenue(tabla_ventas)           # devuelve 2 3
```

```
# realiza una prueba de chi-cuadrado
```

```
resumen(tabla_ventas)
```

Número de casos en la tabla: 10000 Número de

factores: 2

Pruebe la independencia de todos los factores:

```
Chisq = 1,516, gl = 2, valor p = 0,4686
```

Según los recuentos observados en la tabla, el resumen() La función realiza una prueba de chi-cuadrado sobre la independencia de los dos factores. Porque el reportado *p*-valor es mayor que 0.05, la supuesta independencia de los dos factores no se rechaza. Prueba de hipótesis y *p*-valor. Los valores se tratan con más detalle más adelante en este capítulo. A continuación, se examina la aplicación de estadística descriptiva en R.

3.1.4 Estadística descriptiva

Ya se ha demostrado que el resumen() La función proporciona varias estadísticas descriptivas, como la mediana y la mediana, acerca de una variable como la **ventas** marco de datos. Los resultados ahora incluyen los recuentos de los tres niveles del **gastador** variable basada en los ejemplos anteriores que involucran factores.

```
resumen(ventas)
```

cust_id	sales_total	num_of_orders	género	gastador
Min. : 100001	Min. : 30.02	Min. : 1.000	F: 5035	pequeño: 3382
1er Qu.:102501	1er Qu.: 80.29	1er Qu.: 2.000	M: 4965	medio: 5469
Mediana: 105001	Mediana: 151,65	Mediana: 2.000		grande : 1149
Media : 105001	Media : 249,46	Media : 2.428		
3er Qu.:107500	3er Qu.: 295.50	3er Qu.: 3.000		
Max. : 110000	Max. : 7606.09	Max. : 22.000		

El siguiente código proporciona algunas funciones R comunes que incluyen estadísticas descriptivas. Entre paréntesis, los comentarios describen las funciones.

```
# para simplificar las llamadas a funciones, asigne
x <- ventas $ sales_total
y <- ventas $ num_of_orders

cor (x, y)                                # devuelve 0,7508015 (correlación)
cov (x, y)                                 # devuelve 345.2111 (covarianza)
IQR (x)                                    # devuelve 215,21 (rango intercuartílico)
media (x)                                   # devuelve 249,4557 (media)
mediana (x)                                # devuelve 151,65 (mediana)
rango (x)                                    # devuelve 30.02 7606.09 (min max)
sd (x)                                       # devuelve 319.0508 (std. dev.)
var (x)                                      # devuelve 101793.4 (variación)
```

los IQR () La función proporciona la diferencia entre el tercer y el primer cuartil. Las otras funciones se explican por sí mismas por sus nombres. Se anima al lector a revisar los archivos de ayuda disponibles para obtener entradas aceptables y posibles opciones.

La función aplicar() es útil cuando se va a aplicar la misma función a varias variables en un marco de datos. Por ejemplo, el siguiente código R calcula la desviación estándar para las tres primeras variables en **ventas**. En el código, configuración MARGEN = 2 especifica que el Dakota del Sur() La función se aplica sobre las columnas. Otras funciones, como lapply () y sapply (), aplicar una función a una lista o vector. Los lectores pueden consultar los archivos de ayuda de R para aprender a utilizar estas funciones.

```
aplicar (ventas [, c (1: 3)], MARGEN = 2, FUN = sd)
  cust_id      sales_total num_of_orders
  2886.895680     319.050782       1.441119
```

Se pueden aplicar estadísticas descriptivas adicionales con funciones definidas por el usuario. El siguiente código R define una función, mi_rango (), para calcular la diferencia entre los valores máximo y mínimo devueltos por rango() función. En general, las funciones definidas por el usuario son útiles para cualquier tarea u operación que deba repetirse con frecuencia. Más información sobre funciones definidas por el usuario está disponible ingresando **ayuda ("función")** en la consola.

```
# construye una función para proporcionar la diferencia entre
# los valores máximo y mínimo
mi_rango <- función (v) {rango (v) [2] - rango (v) [1]} mi_rango (x)
```

7576.07

3.2 Análisis de datos exploratorios

Hasta ahora, este capítulo ha abordado la importación y exportación de datos en R, los tipos de datos básicos y las operaciones y la generación de estadísticas descriptivas. Funciones como resumen() puede ayudar a los analistas a tener una idea fácil de la magnitud y el rango de los datos, pero otros aspectos, como las relaciones lineales y las distribuciones, son más difíciles de ver en las estadísticas descriptivas. Por ejemplo, el siguiente código muestra una vista resumida de un marco de datos. **datos** con dos columnas **X** y **y**. La salida muestra el rango de **X** y **y**, pero no está claro cuál puede ser la relación entre estas dos variables.

resumen (datos)

	X	y
Min.	-1.90483	-2.16545
1er Qu.:	0,66321	1er Qu.: -0,71451
Mediana:	0,09367	Mediana: -0,03797
Media	: 0,02522	Media : -0,02153
3er Qu.:	0,65414	3er Qu. : 0,55738
Max.	: 2,18471	Max. : 1,70199

Una forma útil de detectar patrones y anomalías en los datos es a través del análisis exploratorio de datos con visualización. La visualización ofrece una visión sencilla y holística de los datos que pueden ser difíciles de captar solo a partir de números y resúmenes. Variables *X* y *y* del marco de datos *datos* en cambio, se puede visualizar en un diagrama de dispersión (Figura 3-5), que describe fácilmente la relación entre dos variables. Una faceta importante de los datos iniciales e

relaciones en el

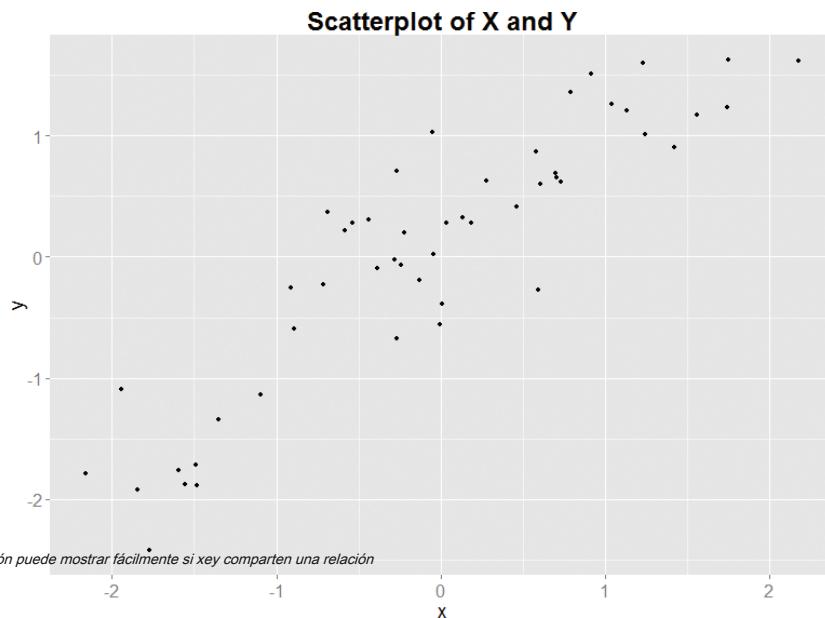


FIGURE 3-5 Un diagrama de dispersión puede mostrar fácilmente si *x* y *y* comparten una relación

El código para generar *datos* así como la Figura 3-5 se muestra a continuación.

```
x <- normal(50)
y <- x + rnorm(50, media = 0, sd = 0.5)

datos <- as.data.frame (cbind (x, y))
```

resumen (datos)

```
biblioteca (ggplot2)
ggplot (datos, aes (x = x, y = y)) +
  geom_point (tamaño = 2) +
  ggtitle ("Diagrama de dispersión de X e Y") + tema (axis.text
  = element_text (tamaño = 12),
  axis.title = element_text (tamaño = 14),
  plot.title = element_text (tamaño = 20, cara = "negrita"))
```

Análisis exploratorio de datos [9] es un enfoque de análisis de datos para revelar las características importantes de un conjunto de datos, principalmente a través de la visualización. Esta sección discute cómo utilizar algunas técnicas básicas de visualización y la función de trazado en R para realizar análisis de datos exploratorios.

3.2.1 Visualización antes del análisis

Para ilustrar la importancia de visualizar datos, considere el cuarteto de Anscombe. El cuarteto de Anscombe consta de cuatro conjuntos de datos, como para demostrar th

FIGURE 3-6 Cuarteto de Anscombe

Los cuatro conjuntos de datos del cuarteto de Anscombe tienen propiedades estadísticas casi idénticas, como se muestra en la Tabla 3-3.

T PODER 3-3 Propiedades estadísticas del cuarteto de Anscombe

Significado de X	9
Varianza de y	11
Significado de y	7.50 (a 2 puntos decimales)

Varianza de y	4.12 o 4.13 (a 2 puntos decimales)
Correlaciones entre X y y	0,816
Línea de regresión lineal	$y = 3,00 + 0,50 X$ (a 2 puntos decimales)

Con base en las propiedades estadísticas casi idénticas en cada conjunto de datos, se podría concluir que estos cuatro conjuntos de datos son bastante similares. Sin embargo, los diagramas de dispersión de la Figura 3-7 cuentan una historia diferente. Cada conjunto de datos se traza como un diagrama de dispersión, y las líneas ajustadas son el resultado de aplicar modelos de regresión lineal. La línea de regresión estimada se ajusta razonablemente bien al conjunto de datos 1. El conjunto de datos 2 es definitivamente no lineal. El conjunto de datos 3 presenta una tendencia lineal, con un valor atípico aparente en $x = 13$. Para el conjunto de datos 4, la línea de regresión se ajusta bastante bien al conjunto de datos. Sin embargo, wi

apropiado.

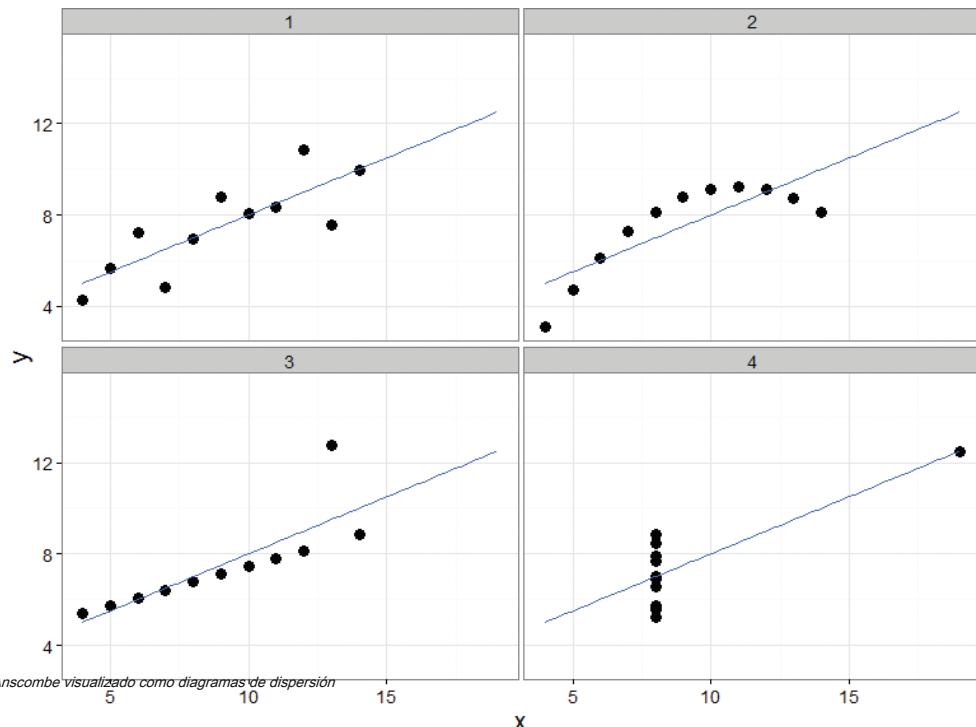


FIGURE 3-7 El cuarteto de Anscombe visualizado como diagramas de dispersión

A continuación se muestra el código R para generar la Figura 3-7. Requiere el paquete R ggplot2 [11], que se puede instalar simplemente ejecutando el comando `install.packages ("ggplot2")`. los anscombe

El conjunto de datos de la gráfica se incluye en la distribución R estándar. Entrar `datos()` para obtener una lista de los conjuntos de datos incluidos en la distribución base R. Entrar `datos(`*Nombre del conjunto de datos*`)` para que un conjunto de datos esté disponible en el espacio de trabajo actual.

En el código que sigue, variable **niveles** se crea utilizando la `gl()` función, que genera factores de cuatro niveles (1, 2, 3 y 4), cada uno repitiéndose 11 veces. Variable **mis datos** se crea utilizando el `con(datos, expresión)` función, que evalúa una **expresión** en un entorno construido a partir de **datos**. En este ejemplo, el **datos** es el anscombe conjunto de datos, que incluye ocho atributos:

x1, x2, x3, x4, y1, y2, y3, y4. los **expresión** parte del código crea un marco de datos a partir del anscombe conjunto de datos, y solo incluye tres atributos: **x, y**, y el grupo al que pertenece cada punto de datos (**mi grupo**).

```
install.packages ("ggplot2") # no es necesario si se ha instalado el paquete
```

datos (anscombe) # cargar el conjunto de datos de anscombe en el espacio de trabajo actual

anscombe

	x1	x2	x3	x4	y1	y2	y3	y4
1	10	10	10	8	8.04	9.14	7.46	6.58
2	8	8	8	8	6.95	8.14	6.77	5.76
3	13	13	13	8	7.58	8.74	12.74	7.71
4	9	9	9	8	8.81	8.77	7.11	8.84
5	11	11	11	8	8.33	9.26	7.81	8.47
6	14	14	14	8	9.96	8.10	8.84	7.04
7	6	6	6	8	7.24	6.13	6.08	5.25
8	4	4	4	19	4.26	3.10	5.39	12.50
9	12	12	12	8	10.84	9.13	8.15	5.56
10	7	7	7	8	4.82	7.26	6.42	7.91
11	5	5	5	8	5.68	4.74	5.73	6.89

```
nrow(anscombe) # número de filas  
[1] 11
```

```
# genera niveles para indicar a qué grupo pertenece cada punto de datos  
niveles <- gl(4, nrow (anscombe)) niveles
```

```
[1] 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3 3 3 3 3 [34] 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4
```

Niveles: 1 2 3 4

Agrupar anscombe en un marco de datos

```
mydata <- con (anscombe, data.frame (x = c (x1, x2, x3, x4), y = c (y1, y2, y3, y4),  
                         mygroup = niveles))
```

mis datos

	x	y mygroup
1	10	8.04
2	8	6,95
3	13	7.58
4	9	8,81

41 19 12,50	4
42 8 5,56	4
43 8 7,91	4
44 8 6,89	4

```
# Hacer diagramas de dispersión usando el paquete ggplot2
biblioteca (ggplot2)
theme_set (theme_bw ()) # establecer el tema del color de la trama

# cree las cuatro gráficas de la Figura 3-7
ggplot (mydata, aes (x, y)) +
  geom_point (tamaño = 4) +
  geom_smooth (método = "lm", fill = NA, fullrange = TRUE) + facet_wrap (~
  mygroup)
```

3.2.2 Datos sucios

Esta sección aborda cómo se pueden detectar datos sucios en la fase de exploración de datos con visualizaciones. En general, los analistas deben buscar anomalías, verificar los datos con conocimiento del dominio y decidir el enfoque más apropiado para limpiar los datos.

Consideré un escenario
retención. figura 3

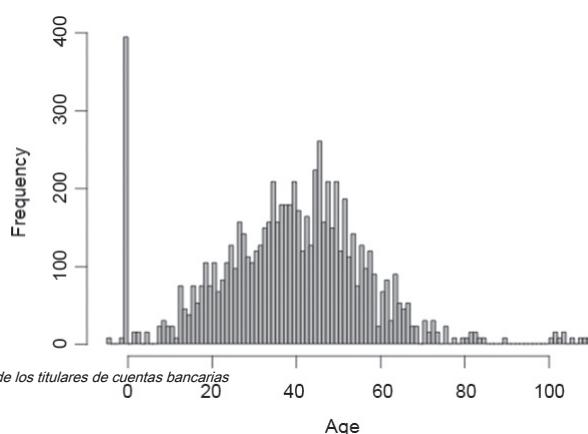


FIGURE 3-8 Distribución por edades de los titulares de cuentas bancarias

Si los datos de edad están en un vector llamado *años*, el gráfico se puede crear con el siguiente script R:

```
hist (edad, pausas = 100, principal = "Distribución por edades de los titulares de cuentas",
      xlab = "Edad", ylab = "Frecuencia", col = "gris")
```

La figura muestra que la edad media de los titulares de la cuenta es de alrededor de 40 años. Algunas cuentas con una edad del titular de la cuenta menor de 10 son inusuales pero plausibles. Pueden ser cuentas de custodia o cuentas de ahorro para la universidad establecidas por los padres de niños pequeños. Estas cuentas deben conservarse para análisis futuros.

Sin embargo, el lado izquierdo del gráfico muestra un gran aumento de clientes que tienen cero años o edades negativas. Es probable que esto sea evidencia de **datos perdidos**. Una posible explicación es que los valores de edad nulos podrían haber sido reemplazados por 0 o valores negativos durante la entrada de datos. Tal ocurrencia puede ser causada por ingresar la edad en un cuadro de texto que solo permite números y no acepta valores vacíos. O puede ser causado por la transferencia de datos entre varios sistemas que tienen diferentes definiciones para valores nulos (como NULL, NA, 0, -1 o -2). Por lo tanto, **limpieza de datos** debe realizarse sobre las cuentas con valores de edad anormales. Los analistas deben examinar más de cerca los registros para decidir si los datos faltantes deben eliminarse o si se puede determinar un valor de edad apropiado utilizando otra información disponible para cada una de las cuentas.

En R, el `is.na()` La función proporciona pruebas para los valores perdidos. El siguiente ejemplo crea un vector `X` donde el cuarto valor no está disponible (N / A). Los `is.na()` devuelve la función CIERTO en cada N / A valor y FALSO de otra manera.

```
x <- c(1, 2, 3, NA, 4) is.na(x)
```

```
[1] FALSO FALSO FALSO VERDADERO FALSO
```

Algunas funciones aritméticas, como `media()`, aplicado a datos que contienen valores perdidos puede producir una N / A resultado. Para evitar esto, configure el `na.rm` parámetro a CIERTO para eliminar el valor faltante durante la ejecución de la función.

```
media(x)
```

```
[1] NA
```

```
media(x, na.rm = VERDADERO)
```

```
[1] 2,5
```

Los `na.exclude()` La función devuelve el objeto sin los casos incompletos.

```
DF <- data.frame(x = c(1, 2, 3), y = c(10, 20, NA)) DF
```

```
xy
```

```
1 1 10
```

```
2 2 20
```

```
3 3 NA
```

```
DF1 <- na.exclude(DF)
```

```
DF1
```

```
xy
```

```
1 1 10
```

```
2 2 20
```

Los titulares de cuentas mayores de 100 años pueden deberse a datos incorrectos causados por errores tipográficos. Otra posibilidad es que estas cuentas se hayan transmitido a los herederos de los titulares de las cuentas originales sin actualizarse. En este caso, es necesario examinar más a fondo los datos y realizar una limpieza de datos si es necesario. Los datos sucios podrían simplemente eliminarse o filtrarse con un umbral de edad para análisis futuros. Si eliminar registros no es una opción, los analistas pueden buscar patrones dentro de los datos y desarrollar un conjunto de heurísticas para atacar el problema de los datos sucios. Por ejemplo, los valores de edad incorrectos podrían reemplazarse con **aproximación** basado en el vecino más cercano — el registro que es más similar al registro en cuestión basado en el análisis de las diferencias en todas las otras variables además de la edad.

La figura 3-9 presenta otro ejemplo de datos sucios. La distribución que se muestra aquí corresponde a la antigüedad de las hipotecas en la cartera de préstamos hipotecarios de un banco. La edad de la hipoteca se calcula restando la fecha de originación del loan número de hipotecas en cada hipoteca ag

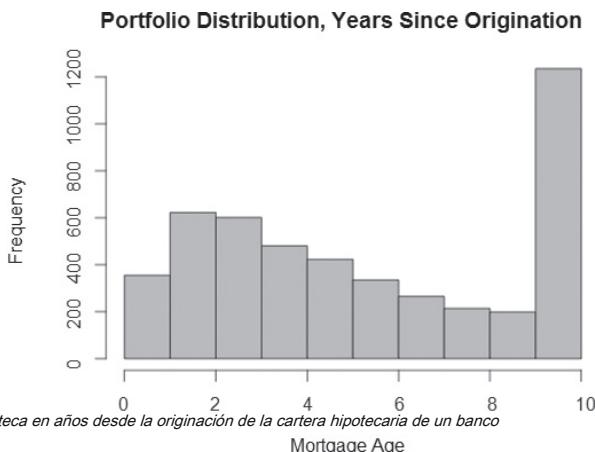


FIGURE 3-9 Distribución de la hipoteca en años desde la originación de la cartera hipotecaria de un banco

Si los datos están en un vector llamado **hipoteca**, La figura 3-9 puede producirse con el siguiente script R.

```
hist(hipoteca, cortes = 10, xlab = "Edad de la hipoteca", col = "gris",
     main = "Distribución de la cartera, años desde su origen")
```

La Figura 3-9 muestra que los préstamos no tienen más de 10 años y estos préstamos a 10 años tienen una frecuencia desproporcionada en comparación con el resto de la población. Una posible explicación es que los préstamos de 10 años no solo incluyen los préstamos que se originaron hace 10 años, sino también los que se originaron antes. En otras palabras, el 10 en el X- eje en realidad significa ≥ 10 . Esto sucede a veces cuando los datos se transfieren de un sistema a otro o porque el proveedor de datos decidió, por alguna razón, no distinguir los préstamos que tienen más de 10 años. Los analistas deben estudiar más los datos y decidir la forma más adecuada de realizar la limpieza de datos.

Los analistas de datos deben realizar verificaciones de cordura contra el conocimiento del dominio y decidir si los datos sucios deben eliminarse. Considere la tarea de averiguar la probabilidad de incumplimiento del préstamo hipotecario. Si las observaciones pasadas sugieren que la mayoría de los incumplimientos ocurren antes del cuarto año y las hipotecas a los 10 años rara vez incumplen, puede ser seguro eliminar los datos sucios y asumir que los préstamos en mora tienen menos de 10 años. Para otros análisis, puede ser necesario rastrear la fuente y averiguar las verdaderas fechas de origen.

Los datos sucios pueden ocurrir debido a actos de omisión. En el **ventas** En los datos utilizados al comienzo de este capítulo, se observó que el número mínimo de pedidos era 1 y el monto mínimo de ventas manuales era \$ 30.02. Por lo tanto, existe una gran posibilidad de que el conjunto de datos proporcionado no incluyera los datos de ventas de todos los clientes, solo los clientes que compraron algo durante el año pasado.

3.2.3 Visualización de una sola variable

El uso de representaciones visuales de datos es un sello distintivo de los análisis de datos exploratorios: dejar que los datos hablen a su audiencia en lugar de imponer una interpretación a los datos. *a priori*. Las secciones 3.2.3 y 3.2.4 examinan formas de mostrar datos para ayudar a explicar las distribuciones subyacentes de una sola variable o las relaciones de dos o más variables.

R tiene muchas funciones disponibles para examinar una sola variable. Algunas de estas funciones se enumeran en la Tabla 3-4.

T PODER 3-4 Funciones de ejemplo para visualizar una sola variable

trama(<i>datos</i>)	Diagrama de dispersión donde <i>x</i> es el índice e <i>y</i> es el valor; adecuado para series de larga duración
barplot <i>datos</i>)	Gráfico de barras con barras verticales u horizontales Gráfico
dotchart <i>datos</i>)	de puntos de Cleveland [12]
hist <i>datos</i>)	Histograma
parcela(<i>densidad</i> (<i>datos</i>))	Gráfico de densidad (un histograma continuo) Gráfico de
vástago(<i>datos</i>)	tallo y hojas
alfombra(<i>datos</i>)	Agregue una representación de alfombra (gráfico 1-d) de los datos a un gráfico existente

Dotchart y Barplot

Los gráficos de puntos y de barras representan valores continuos con etiquetas de una variable discreta. Se puede crear un gráfico de puntos en R con la función dotchart (*x*, *etiqueta* = ...), donde *X* es un vector numérico y *etiqueta*

es un vector de etiquetas categóricas para *X*. Se puede crear un diagrama de barras con el barplot (*altura*) función, donde *altura* representa un vector o matriz. La figura 3-10 muestra (a) un diagrama de puntos y (b) un diagrama de barras basado en mtcars conjunto de datos, que incluye el consumo de combustible y 10 aspectos del diseño de automóviles y el rendimiento de 32 automóviles. Este conjunto de datos viene con la distribución R estándar.

Los gráficos de la Figura 3-10 se pueden generar con el siguiente código R.

```
datos (mtcars)
dotchart(mtcars $ mpg, etiquetas = fila.nombres (mtcars), cex = .7,
         main = "Millas por galón (MPG) de modelos de automóvil", xlab = "MPG")

barplot(table (mtcars $ cyl), main = "Distribución de recuentos de cilindros de automóviles",
        xlab = "Número de cilindros")
```

Gráfico de histograma y densidad

La figura 3-11 (a) incluye un histograma de los ingresos familiares. El histograma muestra una clara concentración de ingresos familiares bajos a la izquierda y la cola larga de los ingresos más altos a la derecha.

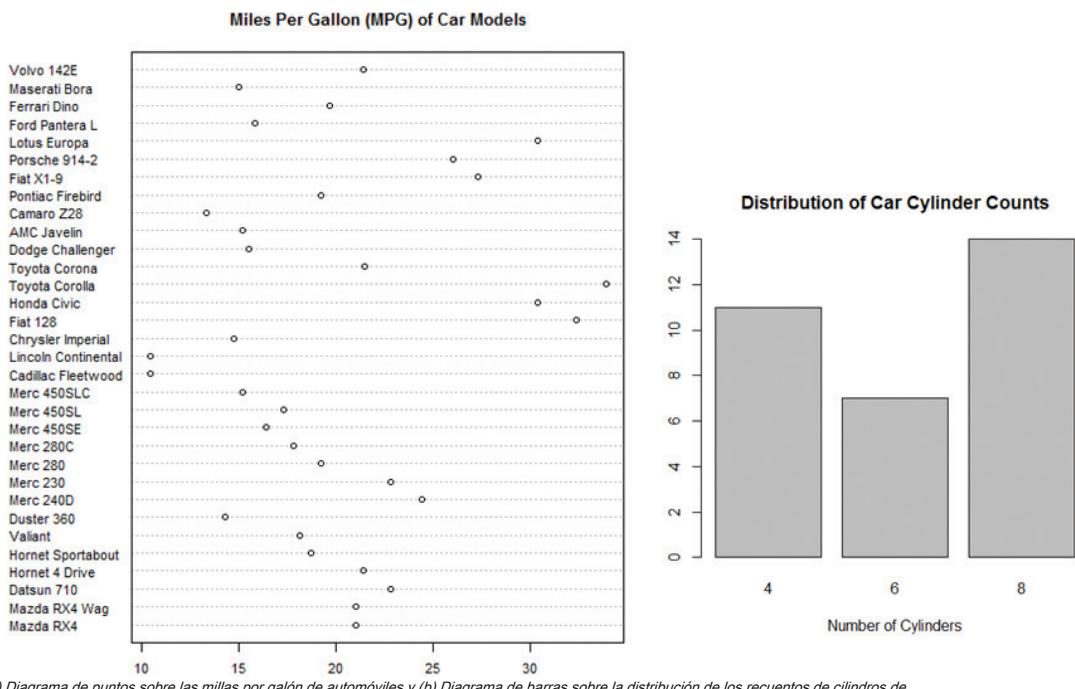


FIGURE 3-10 (a) Diagrama de puntos sobre las millas por galón de automóviles y (b) Diagrama de barras sobre la distribución de los recuentos de cilindros de los automóviles

(a)

(b)

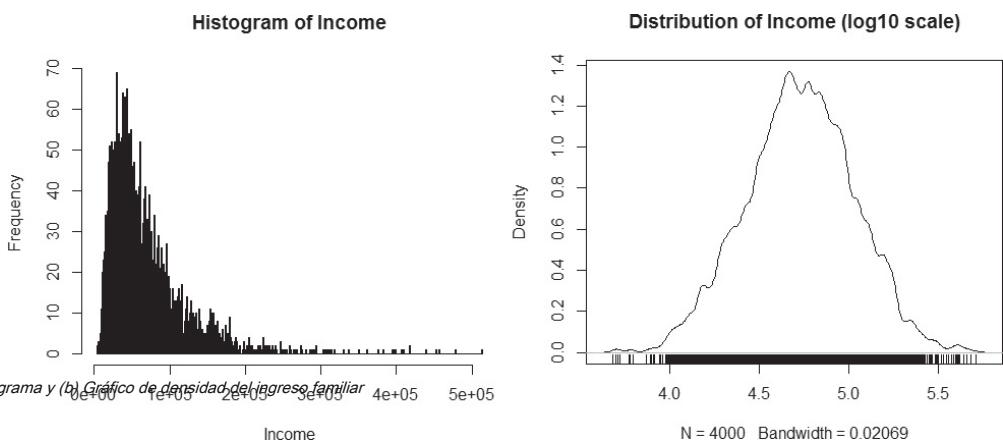


FIGURE 3-11 (a) Histograma y (b) Gráfico de densidad del ingreso familiar

La Figura 3-11 (b) muestra una gráfica de densidad del logaritmo de los valores del ingreso del hogar, que enfatiza la distribución. La distribución del ingreso se concentra en la parte central del gráfico. A continuación, se proporciona el código para generar los dos gráficos de la Figura 3-11. los alfombra() La función crea una gráfica de densidad unidimensional en la parte inferior del gráfico para enfatizar la distribución de la observación.

```
# generar aleatoriamente 4000 observaciones a partir de la distribución logarítmica normal
ingresos <- rlnorm (4000, meanlog = 4, sdlog = 0.7) resumen (ingresos)
```

	Min. 1er Qu.	Mediana	Media 3 ^a Qu.	Max.
	4.301 33.720	54.970 70.320	88.800 659.800	
ingresos <- 1000 * i		bienvenido		
resumen (ingresos)				
	Min. 1er Qu.	Mediana	Media 3 ^a Qu.	Max.
	4301 33720	54970 70320	88800 659800	
# trazar el histo		gramo		
hist (ingresos, bre		aks = 500, xlab = "Ingresos", main = "Histograma de ingresos")		
# parcela de densidad				
plot (densidad (log	10 (ingresos), ajustar = 0,5),			
main = "Distr	Contribución de la renta (escala log10) ")			
# agregue alfombra al	gráfico de densidad			
alfombra (log10 (ingresos))			

En la fase de preparación de datos del ciclo de vida de análisis de datos, se puede obtener el rango y la distribución de los datos. Si los datos están sesgados, ver el logaritmo de los datos (si todo es positivo) puede ayudar a detectar estructuras que de otro modo podrían pasarse por alto en un gráfico con una escala regular no logarítmica.

Al preparar los datos, se deben buscar signos de datos sucios, como se explicó en la sección anterior. Examinar si los datos son unimodales o multimodales dará una idea de cómo muchas poblaciones distintas con diferentes patrones de comportamiento podrían mezclarse en la población general. Muchas técnicas de modelado asumen que los datos siguen una distribución normal. Por lo tanto, es importante saber si el conjunto de datos disponible puede coincidir con ese supuesto antes de aplicar cualquiera de esas técnicas de modelado.

Considera una gráfica de densidad de los precios de los diamantes (en USD). La Figura 3-12 (a) contiene dos diagramas de densidad para cortes de diamantes premium e ideales. El grupo de cortes premium se muestra en rojo y el grupo de cortes ideales se muestra en azul. El rango de precios de los diamantes es amplio; en este caso, oscila entre alrededor de \$ 300 y casi \$ 20 000. Los valores extremos son típicos de datos monetarios como ingresos, valor del cliente, obligaciones tributarias y tamaños de cuentas bancarias.

La Figura 3-12 (b) muestra más detalles de los precios de los diamantes que la Figura 3-12 (a) tomando el logaritmo. Las dos jorobas en el corte premium representan dos grupos distintos de precios de diamantes: Un grupo se centra en iniciar sesión $\text{precio} = 2.9$ (donde el precio es de aproximadamente \$ 794), y los otros centros alrededor del registro $\text{precio} = 3.7$ (donde el precio es de aproximadamente \$ 5012). El corte ideal contiene tres jorobas, centradas alrededor de 2.9, 3.3 y 3.7 respectivamente.

A continuación se muestra el script R para generar los gráficos de la Figura 3-12. los **diamantes** el conjunto de datos viene con el ggplot2 paquete.

```
biblioteca ("ggplot2")
datos (diamantes) # cargar el conjunto de datos de diamantes de ggplot2

# Conserve únicamente los cortes de diamantes premium e ideales
```

```

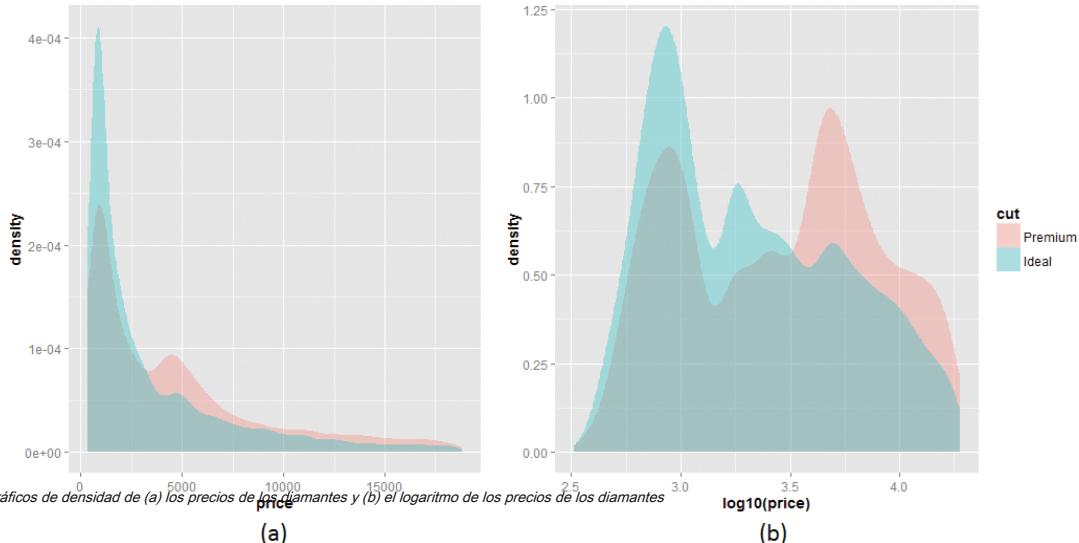
niceDiamonds <- diamantes [diamantes $ cut == "Premium" |
                           diamantes $ cut == "Ideal",]

resumen (niceDiamonds $ cut)
Justa      Buena muy buena      Prima      Ideal
0           0           0       13791     21551

# gráfico de densidad de parcela de precios de diamantes
ggplot (niceDiamonds, aes (x = precio, relleno = corte)) +
  densidad_geom (alpha = .3, color = NA)

# gráfica de densidad de parcela del log10 de los precios de los diamantes
ggplot (niceDiamonds, aes (x = log10 (precio), fill = cut)) +
  densidad_geom (alpha = .3, color = NA)

```



3.2.4 Examen de múltiples variables

Un diagrama de dispersión (mostrado anteriormente en la Figura 3-1 y la Figura 3-5) es una visualización simple y ampliamente utilizada para encontrar la relación entre múltiples variables. Una gráfica de dispersión puede representar datos con hasta cinco variables usando el eje x, el eje y, el tamaño, el color y la forma. Pero generalmente solo se representan de dos a cuatro variables en un diagrama de dispersión para minimizar la confusión. Al examinar un diagrama de dispersión, es necesario prestar mucha atención

a la posible relación entre las variables. Si la relación funcional entre las variables es algo pronunciada, los datos pueden estar aproximadamente a lo largo de una línea recta, una parábola o una curva exponencial. Si variable y está relacionado exponencialmente con X , entonces la trama de X versus Iniciar sesión(y) es aproximadamente lineal. Si el gráfico se parece más a un grupo sin patrón, las variables correspondientes pueden tener una relación débil.

La gráfica de dispersión de la Figura 3-13 muestra la relación de dos variables: X y y . La línea roja que se muestra en el gráfico es la línea ajustada de la regresión lineal. La regresión lineal se revisará en el Capítulo 6, "Teoría y métodos analíticos avanzados: regresión". La figura 3-13 muestra que la línea de regresión no se ajusta bien a los datos. Este es un caso en el que la regresión lineal no puede modelar la relación entre las variables. Métodos alternativos como el loess() La función se puede utilizar para ajustar una línea no lineal a los datos. La curva azul

n lineal

regresión.

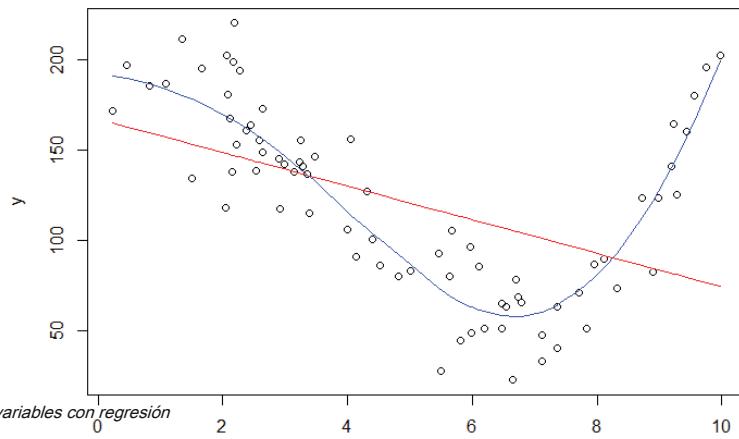


FIGURE 3-13 Examinando dos variables con regresión

El código R para producir la Figura 3-13 es el siguiente. Los runif (75,0,10) generan 75 números entre 0 y 10 con desviaciones aleatorias y los números se ajustan a la distribución uniforme. Los normal (75,0,20) generan 75 números que se ajustan a la distribución normal, con un promedio igual a 0 y la desviación estándar igual a 20. La puntos() La función es una función genérica que dibuja una secuencia de puntos en las coordenadas especificadas. Parámetro tipo = "l" le dice a la función que dibuje una línea continua. Los columnas El parámetro establece el color de la línea, donde 2 representa el color rojo y 4 representa el color azul.

```
# 75 números entre 0 y 10 de distribución uniforme
x <- runif (75, 0, 10)
```

```
x <- ordenar (x)
y <- 200 + x ^ 3 - 10 * x ^ 2 + x + rnorm (75, 0, 20)
```

```
lr <- lm (y ~ x) # regresión lineal
poli <- loess (y ~ x) # LOESS
```

```
ajustar <- predecir (poli) # ajustar una línea no lineal
```

```
trama (x, y)
```

```
# dibuja la línea ajustada para la regresión lineal
```

```
puntos (x, lr $ coeficientes [1] + lr $ coeficientes [2] * x,  
       tipo = "l", col = 2)
```

```
# dibuja la línea ajustada con LOESS
```

```
puntos (x, fit, type = "l", col = 4)
```

Dotchart y Barplot

El gráfico de puntos y el gráfico de barras de la sección anterior pueden visualizar múltiples variables. Ambos utilizan el color como una dimensión adicional para visualizar los datos.

Por lo mismo mtcars conjunto de datos, la Figura 3-14 muestra un gráfico de puntos que agrupa los cilindros del vehículo en el eje y

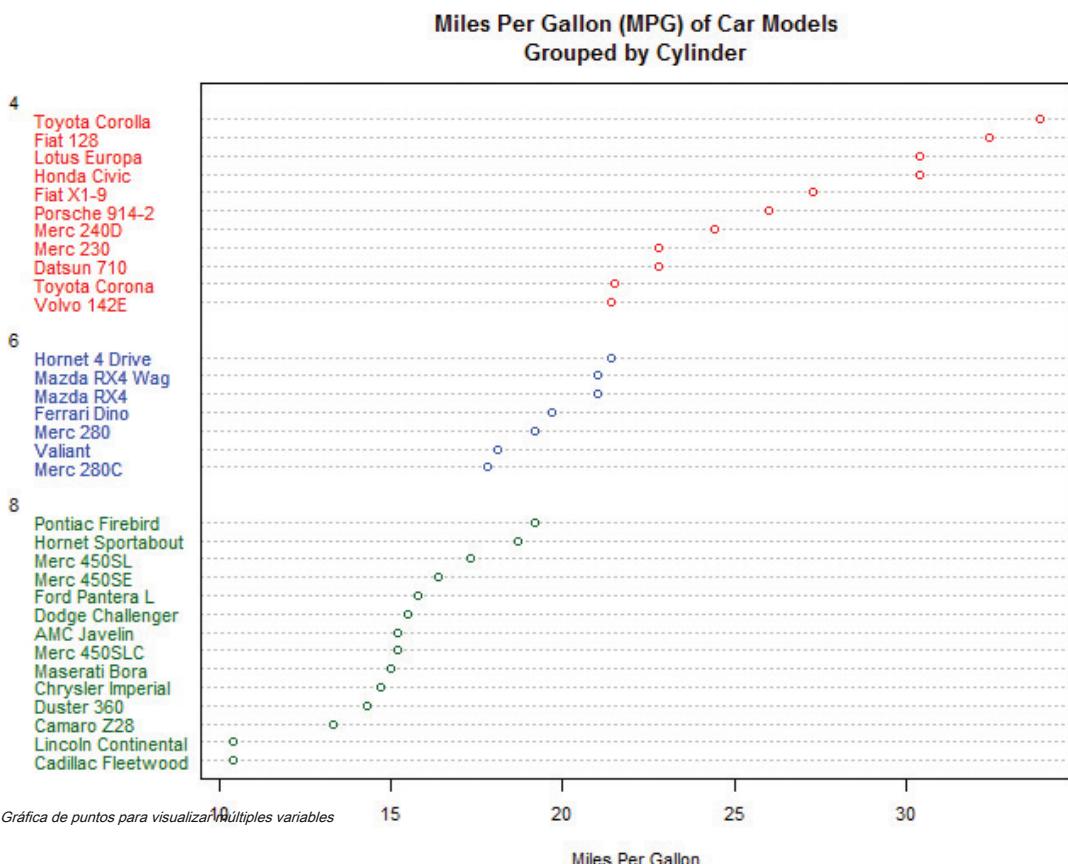


FIGURE 3-14 Gráfica de puntos para visualizar múltiples variables

```
# ordenar por mpg
coches <- mtcars [pedido (mtcars $ mpg).]

# la variable de agrupación debe ser un factor
coches $ cyl <- factor (coches $ cyl)

coches $ color [coches $ cyl == 4] <- "rojo"
coches $ color [coches $ cyl == 6] <- "azul"
coches $ color [coches $ cyl == 8] <- "verde oscuro"

dotchart (coches $ mpg, etiquetas = fila.nombres (coches), cex = .7, grupos = coches $ cyl,
main = "Millas por galón (MPG) de modelos de automóviles \ nGrupos por cilindro", xlab = "Millas por
galón", color = automóviles $ color, gcolor = "negro")
```

El diagrama de barras de la Figura 3-15 visualiza la distribución de los recuentos de cilindros del automóvil y el número de marchas. El representante del eje x generar

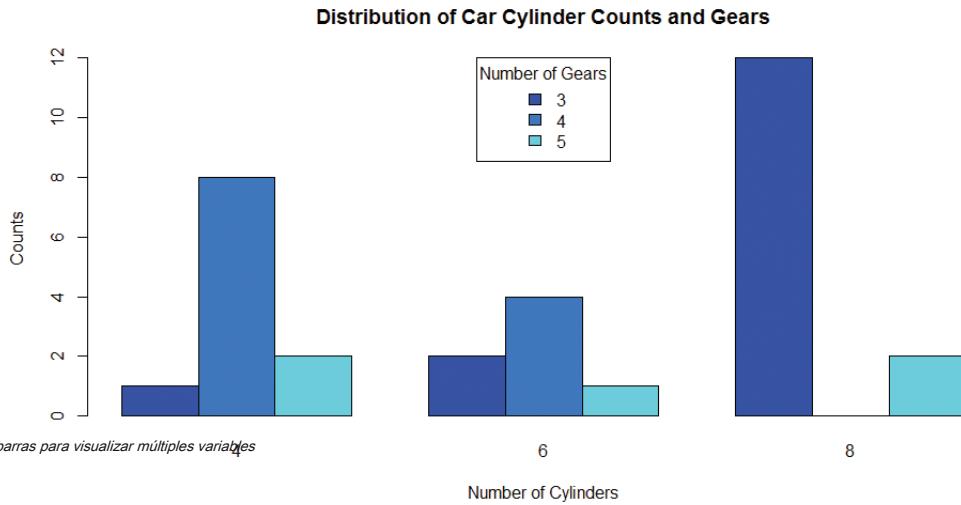


FIGURE 3-15 Gráfica de barras para visualizar múltiples variables

```
cuenta <- tabla (mtcars $ engranaje, mtcars $ cyl)
barplot (count, main = "Distribución de recuentos y engranajes de cilindros del automóvil",
xlab = "Número de cilindros", ylab = "Recuentos", col = c ("#0000FFFF", "#0080FFFF", "#00FFFFFF"),
leyenda = nombres de filas (recuentos), al lado de = VERDADERO,
args.legend = list (x = "top", title = "Number of Gears"))
```

Trama de caja y bigotes

Los diagramas de caja y bigotes muestran la distribución de una variable continua para cada valor de una variable discreta. El diagrama de caja y bigotes de la figura 3-16 visualiza los ingresos medios de los hogares en función de la región de Estados Unidos. El primer dígito del código postal de EE. UU. ("ZIP") corresponde a una región geográfica de los Estados Unidos. En la Figura 3-16, cada punto de datos corresponde al ingreso familiar promedio de un código postal en particular. El eje horizontal representa el primer dígito de un código postal, que va del 0 al 9, donde 0 corresponde a la región noreste de los Estados Unidos (como Maine, Vermont y Massachusetts) y 9 corresponde a la región suroeste (como California y Hawaii). El eje vertical representa el logaritmo de m

de la casa mala

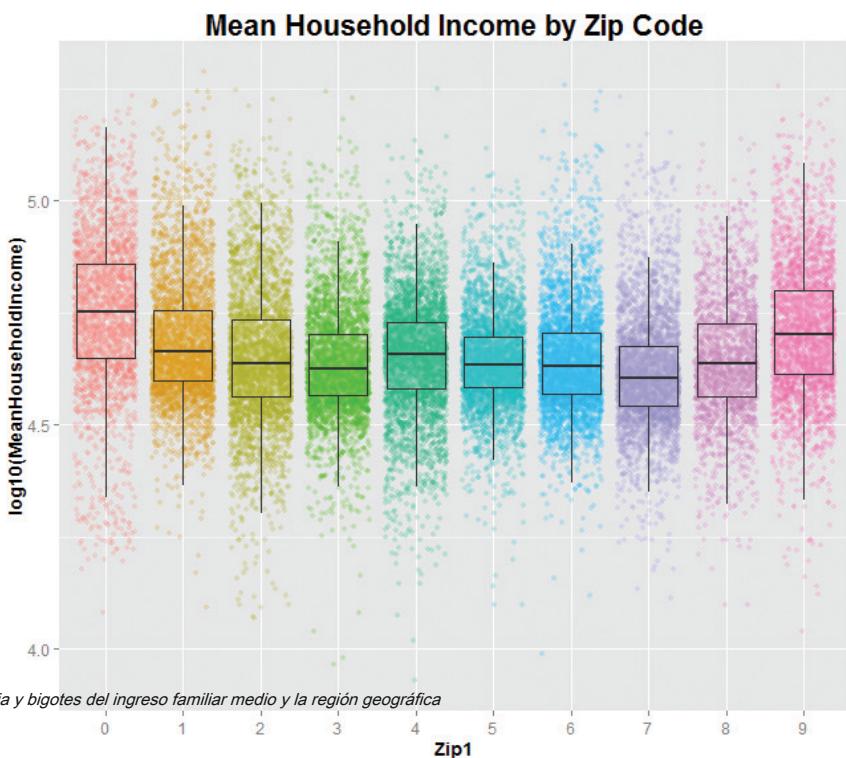


FIGURE 3-16 Un diagrama de caja y bigotes del ingreso familiar medio y la región geográfica

En esta figura, el diagrama de dispersión se muestra debajo del diagrama de caja y bigotes, con algunas fluctuaciones para los puntos de superposición, de modo que cada línea de puntos se ensancha en una franja. La "caja" de la caja y el bigote muestra el rango que contiene el 50% central de los datos, y la línea dentro de la caja es la ubicación del valor mediano. Las bisagras superior e inferior de las casillas corresponden al primer y tercer cuartiles de los datos. El bigote superior se extiende desde la bisagra hasta el valor más alto que se encuentra dentro de $1.5 * \text{IQR}$ de la bisagra. El bigote inferior se extiende desde la bisagra hasta el valor más bajo dentro de $1.5 * \text{IQR}$ de la bisagra. IQR es el rango intercuartil, como se discutió en la Sección 3.1.4. Los puntos fuera de los bigotes pueden considerarse posibles valores atípicos.

El gráfico muestra cómo varía el ingreso familiar por región. Los ingresos medios más altos se encuentran en la región 0 y la región 9. La región 0 es un poco más alta, pero los cuadros de las dos regiones se superponen lo suficiente como para que la diferencia entre las dos regiones probablemente no sea significativa. Los ingresos familiares más bajos tienden a estar en la región 7, que incluye estados como Louisiana, Arkansas y Oklahoma.

Suponiendo un marco de datos llamado **DF** contiene dos columnas (*Ingresos medios del hogar Zip1*), el siguiente script R usa el ggplot2 library [11] para trazar un gráfico similar al de la Figura 3-16.

biblioteca (ggplot2)

```
# trazar el diagrama de dispersión con jittered w / boxplot
# puntos de código de color con códigos postales
# the outlier.size = 0 evita que el diagrama de caja represente el valor atípico
ggplot (data = DF, aes (x = as.factor (Zip1), y = log10 (MeanHouseholdIncome))) +
  geom_point (aes (color = factor (Zip1)), alpha = 0.2, position = "jitter") + geom_boxplot (outlier.size = 0,
  alpha = 0.1) +
  guías (color = FALSO) +
  ggtitle ("Ingresos familiares medios por código postal")
```

Alternativamente, se puede crear un diagrama simple de caja y bigotes con el diagrama de caja () función proporcionada por el paquete base R.

Hexbinplot para grandes conjuntos de datos

Este capítulo ha demostrado que el diagrama de dispersión como visualización popular puede visualizar datos que contienen una o más variables. Pero uno debe tener cuidado al usarlo en datos de gran volumen. Si hay demasiados datos, la estructura de los datos puede volverse difícil de ver en un diagrama de dispersión. Considere un caso para comparar el logaritmo del ingreso familiar con los años de educación, como se muestra en la Figura 3-17. El grupo de la gráfica de dispersión de la izquierda (a) sugiere una relación algo lineal de las dos variables. Sin embargo, uno no puede realmente ver

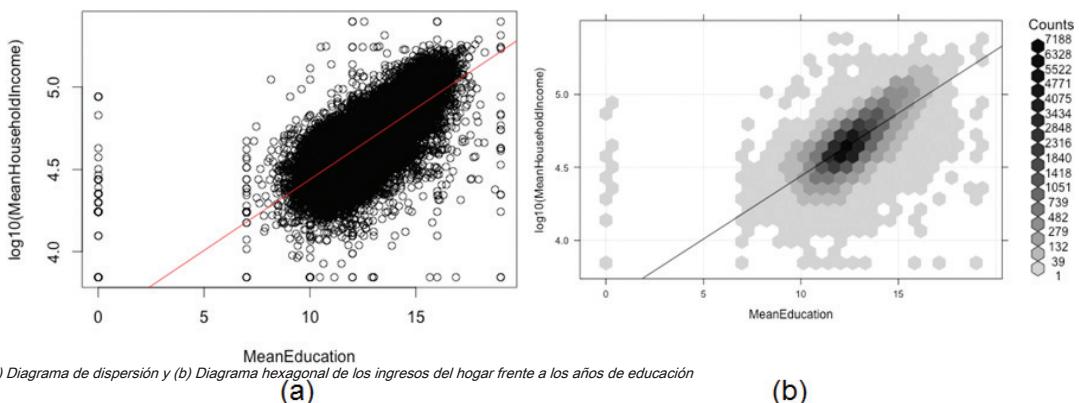


FIGURE 3-17 (a) Diagrama de dispersión y (b) Diagrama hexagonal de los ingresos del hogar frente a los años de educación

Aunque el color y la transparencia se pueden usar en un diagrama de dispersión para abordar este problema, un diagrama hexbin a veces es una mejor alternativa. Un diagrama de hexágono combina las ideas de diagrama de dispersión e histograma. Similar a un diagrama de dispersión, un diagrama hexbin visualiza datos en el *X*-eje y *y*-eje. Los datos se colocan en hexbins y la tercera dimensión usa sombreado para representar la concentración de datos en cada hexbin.

En la Figura 3-17 (b), se trazan los mismos datos usando un diagrama hexbin. El hexbinplot muestra que los datos están agrupados más densamente en una racha que atraviesa el centro del grupo, aproximadamente a lo largo de la línea de regresión. La mayor concentración es de alrededor de 12 años de educación, que se extiende a unos 15 años.

En la Figura 3-17, observe los datos atípicos en *MeanEducation* = 0. Estos puntos de datos pueden corresponder a algunos datos faltantes que necesitan una mayor limpieza.

Asumiendo las dos variables *Ingresos medios del hogar* y *MeanEducation* son de un marco de datos llamado *zcta*, el diagrama de dispersión de la Figura 3-17 (a) se representa mediante el siguiente código R.

```
# trazar los puntos de datos
plot(log10(ingreso medio del hogar) ~ educación media, datos = zcta)
# agregue una línea recta ajustada de la regresión lineal
abline(lm(log10(ingresos medios del hogar) ~ Educación media, datos = zcta), col = 'rojo')
```

Utilizando la *zcta* marco de datos, el diagrama hexbinplot de la Figura 3-17 (b) se traza mediante el siguiente código R. Ejecutar el código requiere el uso de hexbin paquete, que se puede instalar ejecutando Instalar en pc . paquetes ("hexbin").

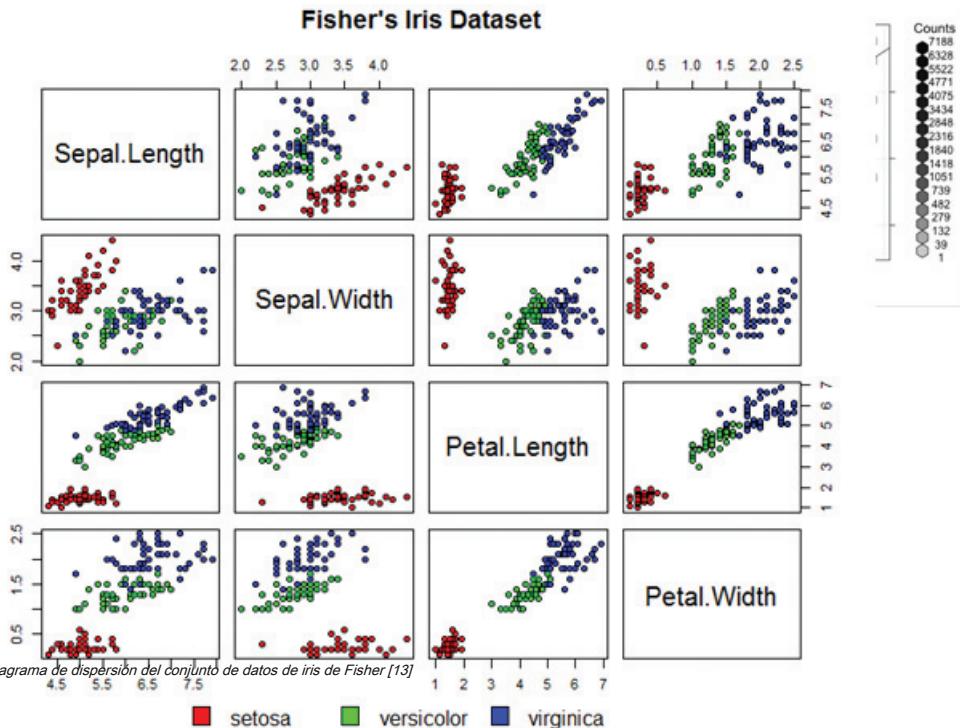
```
biblioteca(hexbin)
# "g" agrega la cuadrícula, "r" agrega la línea de regresión
# sqrt transform on the count da más rango dinámico al sombreado
# inv proporciona la función de transformación inversa de trans
hexbinplot(log10(MeanHouseholdIncome) ~ MeanEducation,
           data = zcta, trans = sqrt, inv = function(x) x ^ 2, type = c("g", "r"))
```

Matriz de gráficos de dispersión

Una matriz de diagramas de dispersión muestra muchos diagramas de dispersión de forma compacta, una al lado de la otra. La matriz del diagrama de dispersión, por lo tanto, puede representar visualmente múltiples atributos de un conjunto de datos para explorar sus relaciones, ampliar las diferencias y revelar patrones ocultos.

Fisher's *iris* El conjunto de datos [13] incluye medidas en centímetros de la longitud del sépalo, el ancho del sépalo, la longitud del pétalo y el ancho del pétalo para 50 flores de tres especies de iris. Las tres especies son *setosa*, *versicolor*, y *virginica*. El conjunto de datos de iris viene con la distribución R estándar.

En la Figura 3-18, todas las variables del conjunto de datos del iris de Fisher (longitud del sépalo, ancho del sépalo, longitud del pétalo y ancho del pétalo) se comparan en una matriz de diagramas de dispersión. Los tres colores diferentes representan tres especies de flores de iris. La matriz de diagramas de dispersión de la Figura 3-18 permite a sus espectadores comparar las diferencias entre las especies de iris para cualquier par de atributos.



Considere la gráfica de dispersión de la primera fila y la tercera columna de la Figura 3-18, donde la longitud del sépalo se compara con la longitud del pétalo. El eje horizontal es la longitud del pétalo y el eje vertical es la longitud del sépalo. El diagrama de dispersión muestra que *versicolor* y *virginica* comparten longitudes similares de sépalos y pétalos, aunque este último tiene pétalos más largos. La longitud de los pétalos de todos *setosa* son aproximadamente iguales, y las longitudes de los pétalos son notablemente más cortas que las de las otras dos especies. El diagrama de dispersión muestra que para *versicolor* y *virginica*, la longitud del sépalo crece linealmente con la longitud del pétalo.

A continuación se proporciona el código R para generar la matriz de gráficos de dispersión.

```
# definir los colores
colores <- c ("rojo", "verde", "azul")

# dibuja la matriz de la trama
pares (iris [1: 4], principal = "Conjunto de datos de iris de Fisher",
       pch = 21, bg = colores [unclass (iris $ Species)])

# establecer parámetro gráfico para recortar el trazado a la región de la figura
par (xpd = VERDADERO)

# agregar leyenda
leyenda (0.2, 0.02, horiz = TRUE, as.vector (unique (iris $ Species)),
       relleno = colores, bty = "n")
```

El vector **colores** de fine el esquema de color del gráfico. Podría cambiarse a algo como colores <- c ("gray50", "blanco", "negro") para hacer los diagramas de dispersión en escala de grises.

Analizar una variable en el tiempo

Visualizar una variable en el tiempo es lo mismo que visualizar cualquier par de variables, pero en este caso el objetivo es identificar patrones específicos de tiempo.

La figura 3-19 traza el número total mensual de pasajeros de líneas aéreas internacionales (en miles) desde enero de 1949 hasta diciembre de 1960. Ingrese trama (AirPassengers) en la consola R para obtener un gráfico similar. La trama muestra que, para cada año, ocurre un gran pico entre julio y agosto, y un pequeño pico ocurre arou

presagio se refiere

como un *estacionalidad e*

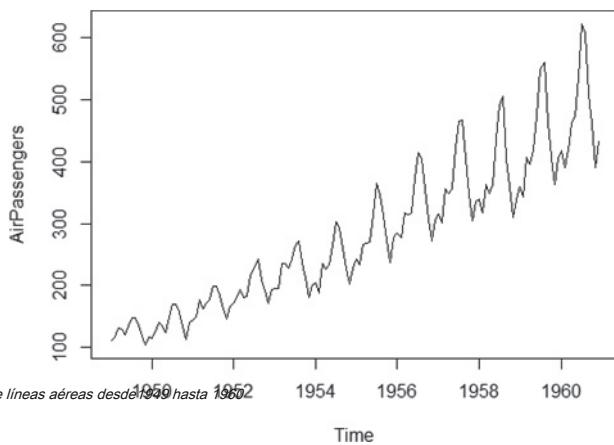


FIGURE 3-19 Número de pasajeros de líneas aéreas desde 1949 hasta 1960

Además, la tendencia general es que el número de pasajeros aéreos aumentó constantemente de 1949 a

1960. El Capítulo 8, “Teoría y métodos analíticos avanzados: análisis de series de tiempo”, analiza el análisis de tales conjuntos de datos con mayor detalle.

3.2.5 Exploración de datos frente a presentación

Usar la visualización para la exploración de datos es diferente a presentar resultados a las partes interesadas. No todos los tipos de tramas son aptas para todos los públicos. La mayoría de los gráficos presentados anteriormente intentan detallar los datos de la manera más clara posible para que los científicos de datos identifiquen estructuras y relaciones. Estos gráficos son de naturaleza más técnica y se adaptan mejor a audiencias técnicas como los científicos de datos. Sin embargo, las partes interesadas sin conocimientos técnicos generalmente prefieren gráficos simples y claros que se centren en el mensaje en lugar de en los datos.

La figura 3-20 muestra la gráfica de densidad de la distribución de los valores de las cuentas de un banco. Los datos han sido convertido al registro 10 escala. El gráfico incluye una alfombra en la parte inferior para mostrar la distribución de la variable. Este gráfico es más adecuado para científicos de datos y analistas comerciales porque proporciona información que

puede ser relevante para el análisis posterior. El gráfico muestra que los valores de la cuenta transformada siguen un no aproximado

aproximadamente $\$ 30$

s

).

Distribution of Account Values (log10 scale)

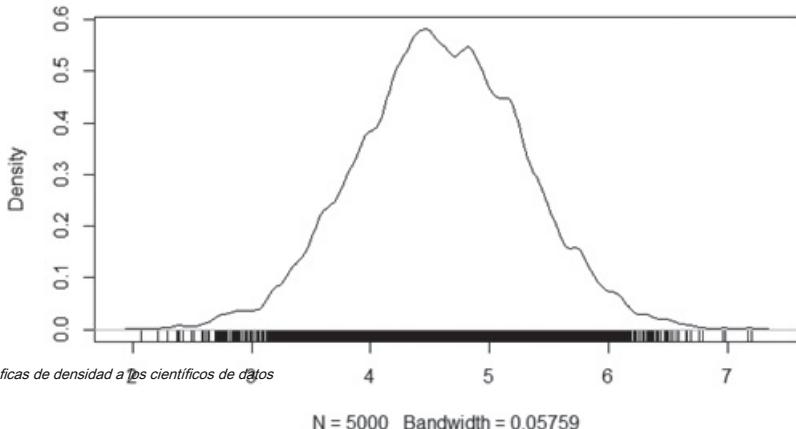


FIGURE 3-20 Es mejor mostrar las gráficas de densidad a los científicos de datos

Las gráficas de densidad son bastante técnicas y contienen tanta información que sería difícil explicar a las partes interesadas menos técnicas. Por ejemplo, sería difícil explicar por qué la cuenta

los valores están en el registro \log escala, y dicha información no es relevante para las partes interesadas. El mismo mensaje se puede transmitir dividiendo los datos en contenedores tipo log y presentándolos como un histograma. Como puede verse en

En la Figura 3-21, la mayor parte de las cuentas están en el rango de $\$ 1,000\text{-}1,000,000$, con la concentración máxima en el rango de $\$ 10\text{-}50K$, que se extiende a $\$ 500K$. Esta representación les da a las partes interesadas una mejor idea de la base de clientes que la gráfica de densidad que se muestra en la Figura 3-20.

Tenga en cuenta que los tamaños de los contenedores deben elegirse cuidadosamente para evitar la distorsión de los datos. En este ejemplo, los contenedores de la Figura 3-21 se eligen en función de las observaciones del gráfico de densidad de la Figura 3-20. Sin el gráfico de densidad, la concentración máxima podría deberse simplemente a las opciones que aparecen algo arbitrarias para los tamaños de los contenedores.

Este sencillo ejemplo aborda las diferentes necesidades de dos grupos de audiencia: analistas y partes interesadas. El Capítulo 12, "El final del juego, o ponerlo todo junto", analiza con más detalle las mejores prácticas para realizar presentaciones a estos dos grupos.

A continuación se muestra el código R para generar los gráficos de la Figura 3-20 y la Figura 3-21.

```
# Generar datos de ingresos normales de registros aleatorios
ingresos = rlnorm (5000, meanlog = log (40000), sdlog = log (5))

# Parte I: Crea la gráfica de densidad
parcela (densidad (log10 (ingresos), ajustar = 0,5),
main = "Distribución de los valores de la cuenta (escala log10)")

# Agregar alfombra a la gráfica de densidad
```

```
alfombra (log10 (ingresos))
```

```
# Parte II: Haz el histograma
# Crear "contenedores de registro"
descansos = c (0, 1000, 5000, 10000, 50000, 100000, 5e5, 1e6, 2e7)
# Cree contenedores y etique los datos
bins = cut (ingresos, cortes, include.lowest = T,
            etiquetas = c ("<1K", "1-5K", "5-10K", "10-50K",
                           "50-100K", "100-500K", "500K-1M", "> 1M"))

# Trazar los contenedores
plot (bins, main = "Distribución de valores de cuenta",
       xlab = "Valor de la cuenta ($ USD)", ylab = "N
```

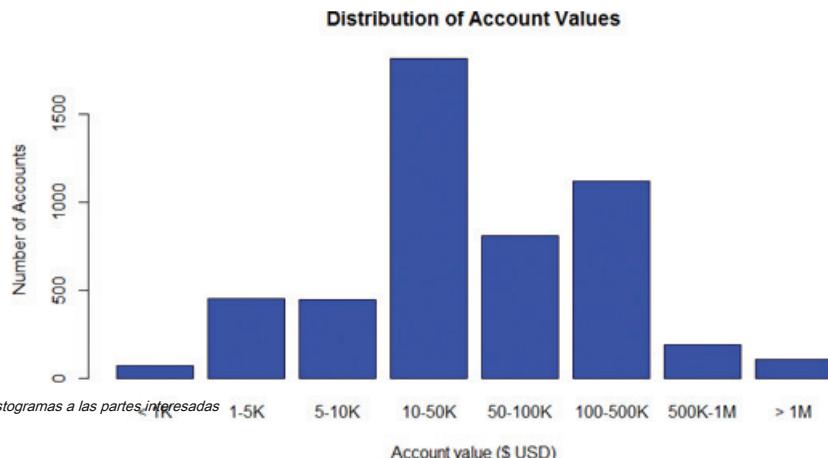


FIGURE 3-21 Es mejor mostrar los histogramas a las partes interesadas

3.3 Métodos estadísticos de evaluación

La visualización es útil para la exploración y presentación de datos, pero las estadísticas son cruciales porque pueden existir durante todo el ciclo de vida de análisis de datos. Las técnicas estadísticas se utilizan durante la exploración y preparación de datos iniciales, la construcción de modelos, la evaluación de los modelos finales y la evaluación de cómo los nuevos modelos mejoran la situación cuando se despliegan en el campo. En particular, las estadísticas pueden ayudar a responder las siguientes preguntas para el análisis de datos:

- Construcción y planificación de modelos
 - ¿Cuáles son las mejores variables de entrada para el modelo?
 - ¿Puede el modelo predecir el resultado dado el input?

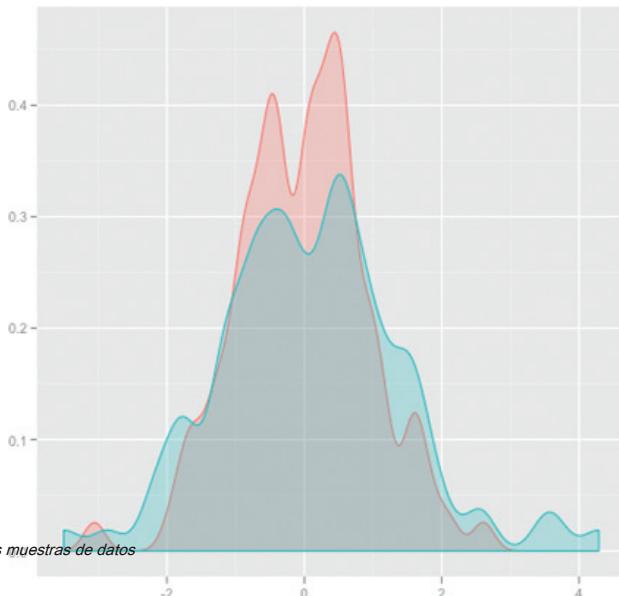
- Evaluación del modelo
 - ¿Es el modelo exacto?
 - ¿El modelo funciona mejor que una suposición obvia?
 - ¿El modelo funciona mejor que otro modelo candidato?
- Despliegue del modelo
 - ¿Es sólida la predicción?
 - ¿Tiene el modelo el efecto deseado (como reducir el costo)?

Esta sección analiza algunas herramientas estadísticas útiles que pueden responder a estas preguntas.

3.3.1 Prueba de hipótesis

Al comparar poblaciones, como probar o evaluar la diferencia de personas a partir de dos muestras de datos (Figura 3-2) e de la diferencia es

prueba de hipótesis



El concepto básico de la prueba de hipótesis es formar una afirmación y probarla con datos. Al realizar pruebas de hipótesis, la suposición común es que no hay diferencia entre dos muestras. Esta suposición se utiliza como la posición predeterminada para construir la prueba o realizar un experimento científico.

Los estadísticos se refieren a esto como el **hipótesis nula (H_0)**, los **hipótesis alternativa (H_{un})** es que hay un

diferencia entre dos muestras. Por ejemplo, si la tarea es identificar el efecto del fármaco A en comparación con el fármaco B en los pacientes, la hipótesis nula y la hipótesis alternativa sería la siguiente.

- H_0 : El fármaco A y el fármaco B tienen el mismo efecto en los pacientes.
- H_{un} : El fármaco A tiene un efecto mayor que el fármaco B en los pacientes.

Si la tarea es identificar si la Campaña de publicidad C es efectiva para reducir la rotación de clientes, la hipótesis nula y la hipótesis alternativa serían las siguientes.

- H_0 : La campaña C no reduce la pérdida de clientes mejor que el método de campaña actual.
- H_{un} : La Campaña C reduce la pérdida de clientes mejor que la campaña actual.

Es importante enunciar la hipótesis nula y la hipótesis alternativa, porque es probable que una declaración incorrecta socave los pasos posteriores del proceso de prueba de hipótesis. Una prueba de hipótesis conduce a rechazar la hipótesis nula a favor de la alternativa o a no rechazar la hipótesis nula.

La tabla 3-5 incluye algunos ejemplos de hipótesis nulas y alternativas que deben responderse durante el ciclo de vida analítico.

T PODER 3-5 Ejemplo de hipótesis nulas e hipótesis alternativas

Pronóstico de precisión	Modelo X <i>no predice</i> mejor que el modelo existente.	Modelo X <i>predice</i> mejor que el modelo existente.
Recomendación Motor	AlgoritmoY <i>no produce</i> mejores recomendaciones que el algoritmo actual usado.	AlgoritmoY <i>produce</i> mejores recomendaciones que el algoritmo actual que se utiliza.
Regresión Modelado	Esta variable <i>no afecta</i> el resultado porque su coeficiente es <i>cero</i> .	Esta variable <i>afecta</i> el resultado porque su coeficiente no es <i>cero</i> .

Una vez que se crea un modelo sobre los datos de entrenamiento, es necesario evaluarlo sobre los datos de prueba para ver si el modelo propuesto predice mejor que el modelo existente que se está utilizando actualmente. La hipótesis nula es que el modelo propuesto no predice mejor que el modelo existente. La hipótesis alternativa es que el modelo propuesto predice mejor que el modelo existente. En el pronóstico de precisión, el modelo nulo podría ser que las ventas del próximo mes sean las mismas que las del mes anterior. La prueba de hipótesis debe evaluar si el modelo propuesto proporciona una mejor predicción. Tome un motor de recomendaciones como ejemplo. La hipótesis nula podría ser que el nuevo algoritmo no produce mejores recomendaciones que el algoritmo actual que se está implementando.

Al evaluar un modelo, a veces es necesario determinar si una determinada variable de entrada mejora el modelo. En el análisis de regresión (Capítulo 6), por ejemplo, esto es lo mismo que preguntar si el coeficiente de regresión de una variable es cero. La hipótesis nula es que el coeficiente es cero, lo que significa que la variable no tiene un impacto en el resultado. La hipótesis alternativa es que el coeficiente es distinto de cero, lo que significa que la variable tiene un impacto en el resultado.

Una prueba de hipótesis común es comparar los medios de dos poblaciones. Dos de estas pruebas de hipótesis se analizan en la Sección

3.3.2.

3.3.2 Diferencia de medias

La prueba de hipótesis es un enfoque común para hacer inferencias sobre si las dos poblaciones, denotadas *popular 1* y *popular 2*, son diferentes entre sí. Esta sección proporciona dos pruebas de hipótesis para comparar las medias de las respectivas poblaciones basadas en muestras extraídas al azar de cada población. Específicamente, las dos pruebas de hipótesis de esta sección consideran las siguientes hipótesis nula y alternativa.

- $H_0: \mu_1 = \mu_2$
- $H_{\text{ALT}}: \mu_1 \neq \mu_2$

los μ_1 y μ_2 denotar las medias poblacionales de *popular 1* y *popular 2*, respectivamente. El enfoque de prueba básico es comparar las medias de la muestra observadas, X_1 y X_2 , correspondiente a cada población. Si los valores de X_1 y X_2 son aproximadamente iguales entre sí, las distribuciones de X_1 y

X_2 se superponen sustancialmente (Figura 3-23) y se apoya la hipótesis nula. Una gran diferencia observada entre la muestra

en los medios puede ser te

Sería rechazado. Formalmente, la diferencia

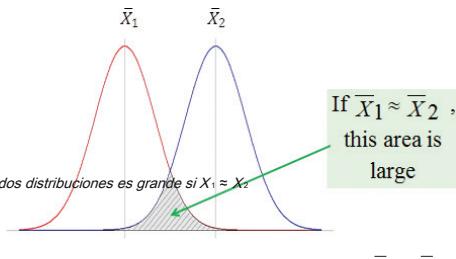


FIGURE 3-23 La superposición de las dos distribuciones es grande si $X_1 \approx X_2$,

Prueba t de Student

Del estudiante *t*- La prueba asume que las distribuciones de las dos poblaciones tienen iguales pero desconocidas variaciones. Suponer *norte₁* y *norte₂*: las muestras se seleccionan de forma aleatoria e independiente de dos poblaciones, *popular 1* y *popular 2*, respectivamente. Si cada población se distribuye normalmente con la misma media ($\mu_1 = \mu_2$) y con la misma varianza, entonces *T* (la estadística *t*), dada en la ecuación 3-1, sigue un **distribución t** con *norte₁ + norte₂ - 2 grados de libertad (gl)*.

$$T = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{\frac{s_{\text{pags}}^2}{norte_1} + \frac{1}{norte_2}}}$$

dónde

$$s_{\text{pags}}^2 = \frac{(norte_1 - 1) s_1^2 + (norte_2 - 1) s_2^2}{norte_1 + norte_2 - 2} \quad (3-1)$$

La forma del t - La distribución es similar a la distribución normal. De hecho, a medida que los grados de libertad se acercan a 30 o más, t - La distribución es casi idéntica a la distribución normal. Porque el numerador de T es la diferencia de las medias muestrales, si el valor observado de T está lo suficientemente lejos de cero como para que la probabilidad de observar tal valor de T sea poco probable, se rechazaría la hipótesis nula de que las medias de la población son iguales. Por lo tanto, para una pequeña probabilidad, digamos $\alpha = 0,05$, T^* se determina de manera que

$P(T \geq T^*) = 0,05$. Una vez recolectadas las muestras y el valor observado de T se calcula de acuerdo con

Ecuación 3-1, la hipótesis nula ($\mu_1 = \mu_2$) es rechazada si $T \geq T^*$.

En la prueba de hipótesis, en general, la pequeña probabilidad, α , es conocido como el **nivel de significación** del examen.

El nivel de significación de la prueba es la probabilidad de rechazar la hipótesis nula, cuando la hipótesis nula es en realidad CIERTO. En otras palabras, para $\alpha = 0,05$, si las medias de las dos poblaciones son realmente iguales, entonces en un muestreo aleatorio repetido, la magnitud observada de T solo excedería T^* 5% del tiempo.

En el siguiente ejemplo de código R, se seleccionan al azar 10 observaciones de dos poblaciones distribuidas normalmente y se asignan a las variables X y y . Las dos poblaciones tienen una media de 100 y 105, respectivamente, y una desviación estándar igual a 5. Student's t -Luego se realiza una prueba para determinar si las muestras aleatorias obtenidas apoyan el rechazo de la hipótesis nula.

```
# generar observaciones aleatorias de las dos poblaciones
x <- rnorm(10, media = 100, sd = 5)           # distribución normal centrada en 100
y <- rnorm(20, media = 105, sd = 5)           # distribución normal centrada en 105
```

```
t.test(x, y, var.equal = TRUE)                 # ejecutar la prueba t de Student
```

Prueba t de dos muestras

```
datos: X y Y
t = -1.7828, gl = 28, valor p = 0.08547
alternar Hipótesis activa: la verdadera diferencia en las medias no es igual al intervalo de confianza de
95 por 0%:
- 6.1 611557 0.4271893
muestra estimados:
media de x media de y
102.21 36 105.0806
```

De la salida R, el valor observado de T es $t = -1.7828$. El signo negativo se debe al hecho de que el media muestral de X es menor que la media muestral de y . Utilizando la `qt()` función en R, a T El valor de 2,0484 corresponde a un nivel de significación de 0,05.

```
# obtener el valor t para una prueba bilateral con un nivel de significancia de 0.05
qt(p = 0.05 / 2, gl = 28, cola inferior = FALSO)
2.048407
```

Porque la magnitud de la observada T la estadística es menor que la T valor correspondiente al nivel de significación 0,05 ($| -1,7828 | < 2,0484$), la hipótesis completa no se rechaza. Porque la hipótesis alternativa es que los hombres no son iguales ($\mu_1 \neq \mu_2$), las probabilidades de ambos $\mu_1 > \mu_2$ y $\mu_1 < \mu_2$ necesita ser considerado. Esta forma de estudiante t -prueba se conoce como **prueba de hipótesis de dos caras**, y es necesario para la suma de los probabilidades bajo ambas colas del t -distribución para igualar el nivel de significación. Se acostumbra uniformemente

divide el nivel de significación entre ambas colas. Entonces, $p = 0.05 / 2 = 0.025$ se utilizó en el `qt()` función para obtener el apropiado t -valor.

Para simplificar la comparación de t -resultados de la prueba al nivel de significancia, la salida R incluye una cantidad conocida como **valor p**. En el ejemplo anterior, el `pvals`- El valor es 0.08547, que es la suma de $P(T \leq -1.7828)$ y $P(T \geq 1.7828)$. La figura 3-24 ilustra el t -estadística para el área debajo de la cola de un t -distribución. Los $-t$ y t son los observados al $P(T \leq -1.7828) = 1.7828$. El área sombreada de la izquierda corresponde al $P(T \geq 1.7828)$.

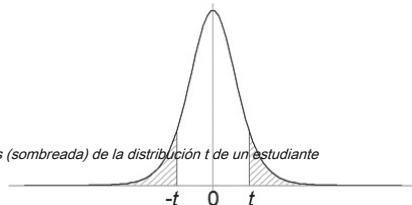


FIGURE 3-24 Área debajo de las colas (sombreada) de la distribución t de un estudiante

En la salida R, para un nivel de significancia de 0.05, la hipótesis nula no sería rechazada porque la probabilidad de una T un valor de magnitud 1,7828 o mayor ocurriría con una probabilidad menor que 0.05. Sin embargo, basado en el `pvals`- valor, si el nivel de significancia se eligiera para ser 0.10, en lugar de 0.05, la hipótesis nula sería rechazada. En general, el `pvals`- El valor ofrece la probabilidad de observar tal resultado muestral dada la hipótesis nula es CIERTO.

Una suposición clave en el uso de Student's t -prueba es que las varianzas poblacionales son iguales. En el ejemplo anterior, el `t.test()` la llamada de función incluye `var.equal = VERDADERO` para especificar que se debe suponer la igualdad de las varianzas. Si esa suposición no es apropiada, entonces Welch t -debe utilizarse la prueba.

Prueba t de Welch

Cuando el supuesto de varianza de la población igual no está justificado al realizar la t -prueba de la diferencia de medias, Welch t -La prueba [14] se puede utilizar basándose en T expresado en la Ecuación 3-2.

$$T_{\text{welch}} = \frac{X_1 - X_2}{\sqrt{\frac{S_{\text{f}+2}^2}{n_1} + \frac{S_2^2}{n_2}}} \quad (3-2)$$

dónde X_1 , $S_{\text{f}+2}$, n_1 y n_2 corresponden al \bar{x}_1 -la media de la muestra, la varianza de la muestra y el tamaño de la muestra. Darse cuenta de que Welch's t -prueba utiliza la varianza de la muestra (S_2) para cada población en lugar de la varianza de la muestra combinada.

En la prueba de Welch, bajo los restantes supuestos de muestras aleatorias de dos poblaciones normales con la misma media, la distribución de T es aproximado por el t -distribución. El siguiente código R realiza el Welch t -prueba en el mismo conjunto de datos analizados en el anterior Student's t -ejemplo de prueba.

```
t.test(x, y, var.equal = FALSE) # ejecutar la prueba t de Welch
```

Prueba t de dos muestras de Welch

```
datos: X y Y
t = -1 . 6596, gl = 15,118, valor p = 0,1176
alternar Hipótesis activa: la verdadera diferencia en las medias no es igual al intervalo de confianza de
95 por 0%:
- 6.5 46629 0,812663
muestra estimados:
media de x media de y
102.21 36 105.0806
```

En este ejemplo particular de uso de Welch *t*-prueba, el *pags*- El valor es 0.1176, que es mayor que el *pags*- valor de 0.08547 observado en el Student's *t*-ejemplo de prueba. En este caso, la hipótesis nula no se rechazaría en un nivel de significancia de 0.10 o 0.05.

Cabe señalar que el cálculo de los grados de libertad no es tan sencillo como en el *t*- prueba. De hecho, el cálculo de los grados de libertad a menudo da como resultado un valor no entero, como en este ejemplo. Los grados de libertad de Welch *t*- La prueba se define en la Ecuación 3-3.

$$df = \frac{\frac{(n_1 - 1)S_{12}^2}{S_{12}^2 + S_{22}^2}}{\frac{(n_2 - 1)S_{22}^2}{S_{12}^2 + S_{22}^2}} \quad (3-3)$$

Tanto en Student's como en Welch's *t*- En los ejemplos de prueba, la salida R proporciona intervalos de confianza del 95% en la diferencia de las medias. En ambos ejemplos, los intervalos de confianza van desde cero. Independientemente del resultado de la prueba de hipótesis, el intervalo de confianza proporciona una estimación de intervalo de la diferencia de las medias de la población, no solo una estimación puntual.

UN *intervalo de confianza* es una estimación de intervalo de un parámetro o característica poblacional basada en datos de muestra. Se utiliza un intervalo de confianza para indicar la incertidumbre de una estimación puntual. Si X es la estimación de alguna media poblacional desconocida μ , el intervalo de confianza proporciona una idea de qué tan cerca X es a lo desconocido μ . Por ejemplo, un intervalo de confianza del 95% para una media poblacional se extiende a CIERTO, pero desconocido significa el 95% de las veces. Considere la Figura 3-25 como ejemplo. Suponga que el nivel de confianza es del 95%. Si la tarea es estimar la media de un valor desconocido μ en una distribución normal con estándar conocido

desviación σ y la estimación basada en n observaciones es X , entonces el intervalo $X \pm 2\sigma$ se extiende a lo largo de lo desconocido valor de μ con un 95% de probabilidad. Si uno toma 100 muestras diferentes y calcula el 95% de con \bar{x}

intervalo de confianza para el hombre, se espera que 95 de los 100 intervalos de confianza se sitúen a caballo entre la media de la población μ .



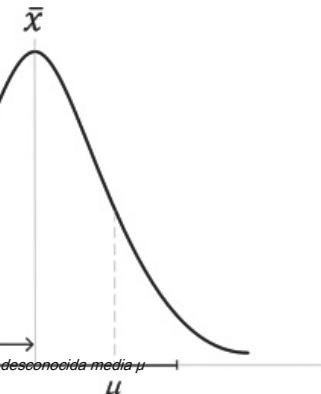


FIGURE 3-25 Un intervalo de confianza del 95% que abarca la población desconocida media μ

Los intervalos de confianza aparecen nuevamente en la Sección 3.3.6 sobre ANOVA. Volviendo a la discusión de la prueba de hipótesis, un supuesto clave tanto en el de Student como en el de Welch *t*- La prueba es que el atributo de población relevante se distribuye normalmente. En el caso de datos que no se distribuyen normalmente, a veces es posible transformar los datos recopilados para aproximarlos a una distribución normal. Por ejemplo, tomar el logaritmo de un conjunto de datos a menudo puede transformar datos sesgados en un conjunto de datos que sea al menos simétrico alrededor de su media. Sin embargo, si tales transformaciones son ineficaces, existen pruebas como la prueba de suma de rangos de Wilcoxon que se pueden aplicar para ver si dos distribuciones de población son diferentes.

3.3.3 Prueba de suma de rangos de Wilcoxon

UN *t*- prueba representa un **prueba paramétrica** en el sentido de que hace suposiciones sobre las distribuciones de población de las que se extraen las muestras. Si no se puede suponer o transformar las poblaciones para que sigan una distribución normal, **prueba no paramétrica** puede ser usado. Los **Prueba de suma de rangos de Wilcoxon** [15] es una prueba de hipótesis no paramétrica que comprueba si dos poblaciones están distribuidas de forma idéntica. Suponiendo que las dos poblaciones están distribuidas de manera idéntica, uno esperaría que el orden de las observaciones muestreadas se mezclaría uniformemente entre sí. Por ejemplo, al ordenar las observaciones, no se esperaría ver un gran número de observaciones de una población agrupada, especialmente al principio o al final del ordenamiento.

Que las dos poblaciones vuelvan a ser *pop1* y *pop2*, con muestras de tamaño aleatorios independientes *norte₁* y *norte₂* respectivamente. El número total de observaciones es entonces $N = n_{1+} norte_2$. El primer paso de la prueba de Wilcoxon es clasificar el conjunto de observaciones de los dos grupos como si procedieran de un grupo grande. El más pequeño La observación recibe un rango de 1, la segunda observación más pequeña recibe un rango de 2, y así sucesivamente, a la observación más grande se le asigna el rango de *NORTE*. Los vínculos entre las observaciones reciben un rango igual al promedio de los rangos que abarcan. La prueba utiliza rangos en lugar de resultados numéricos para evitar suposiciones específicas sobre la forma de la distribución.

Después de clasificar todas las observaciones, los rangos asignados se suman para al menos una muestra de población. Si la distribución de *pop1* se desplaza a la derecha de la otra distribución, la suma de rango correspondiente a *pop1* La muestra debe ser mayor que la suma de rangos de *pop2*. La prueba de suma de rangos de Wilcoxon determina

significado de las sumas de rango observadas. El siguiente código R realiza la prueba en el mismo conjunto de datos utilizado para el anterior *t*-prueba.

```
wilcox.test (x, y, conf.int = TRUE)
```

Prueba de suma de rangos de Wilcoxon

```
datos: X y Y
W = 55 , valor p = 0,04903
alternar Hipótesis activa: el cambio de ubicación real no es igual al intervalo de confianza de
95 por 0%:
 - 6.2 596774 -0,1240618
muestra estimados:
 diff erencia en la ubicación
- 3.417 658
```

los wilcox.test () La función clasifica las observaciones, determina las respectivas sumas de rango correspondientes a la muestra de cada población, y luego determina la probabilidad de que se observen tales sumas de rango de tal magnitud asumiendo que las distribuciones de población son idénticas. En este ejemplo, la probabilidad viene dada por *pags*- valor de 0,04903. Por lo tanto, la hipótesis nula se rechazaría a un nivel de significancia de 0.05. Se advierte al lector que no debe interpretar que una prueba de hipótesis es claramente mejor que otra basada únicamente en los ejemplos que se dan en esta sección.

Debido a que la prueba de Wilcoxon no asume nada sobre la distribución de la población, generalmente se considera más robusta que la *t*-prueba. En otras palabras, hay menos supuestos que violar. Sin embargo, cuando sea razonable suponer que los datos se distribuyen normalmente, Student's o Welch's *t*-prueba es una prueba de hipótesis apropiada a considerar.

3.3.4 Errores tipo I y tipo II

Una prueba de hipótesis puede dar como resultado dos tipos de errores, dependiendo de si la prueba acepta o rechaza la hipótesis nula. Estos dos errores se conocen como errores de tipo I y de tipo II.

- UN **error de tipo I** es el rechazo de la hipótesis nula cuando la hipótesis nula es CIERTO. La probabilidad del error de tipo I se denota con la letra griega α .
- UN **error tipo II** es la aceptación de una hipótesis nula cuando la hipótesis nula es FALSO. La probabilidad del error tipo II se indica con la letra griega β .

La tabla 3-6 enumera los cuatro estados posibles de una prueba de hipótesis, incluidos los dos tipos de errores.

T PODER 3-6 Error tipo I y tipo II

H_0 es aceptado	Resultado correcto	<i>Error tipo II</i>
H_0 se rechaza	<i>Error tipo I</i>	Resultado correcto

El nivel de significación, como se menciona en la *t*-discusión de prueba, es equivalente al error de tipo I.

Para un nivel de significación como $\alpha = 0.05$, si la hipótesis nula ($\mu_1 = \mu_2$) es CIERTO, hay un 5% de probabilidad de que el observado T El valor basado en los datos de la muestra será lo suficientemente grande como para rechazar la hipótesis nula. Por seleccionar-

Con un nivel de significancia apropiado, la probabilidad de cometer un error de tipo I puede definirse antes de que se recopile o analice cualquier dato.

La probabilidad de cometer un error de Tipo II es algo más difícil de determinar. Si dos medias poblacionales realmente no son iguales, la probabilidad de cometer un error de tipo II dependerá de la distancia entre las medias. Para reducir la probabilidad de un error de tipo II a un nivel razonable, a menudo es necesario aumentar el tamaño de la muestra. Este tema se aborda en la siguiente sección.

3.3.5 Potencia y tamaño de la muestra

los **poder** de una prueba es la probabilidad de rechazar correctamente la hipótesis nula. Se denota por $1 - \beta$, donde β es la probabilidad de un error de tipo II. Debido a que la potencia de una prueba mejora a medida que aumenta el tamaño de la muestra, la potencia se utiliza para determinar el tamaño de muestra necesario. En la diferencia de medias, el poder de una prueba de hipótesis depende de la verdadera diferencia de las medias de la población. En otras palabras, para un nivel de significación fijo, se requiere un tamaño de muestra mayor para detectar una diferencia menor en las medias. En general, la magnitud de la diferencia es $k\delta$

effecto tamaño, δ , como ilu

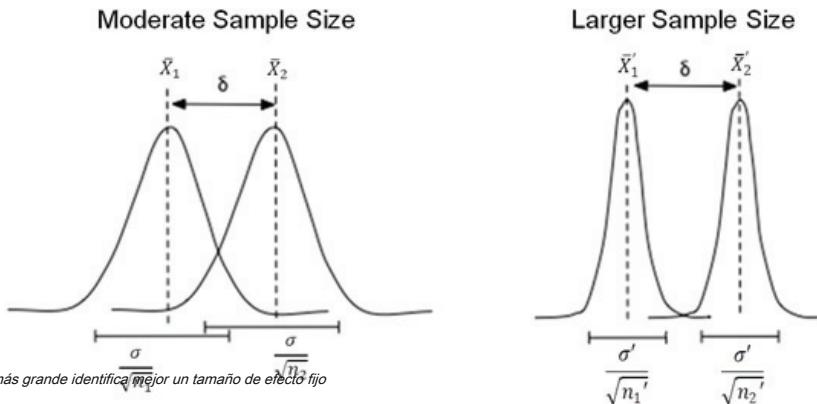


FIGURE 3-26 Un tamaño de muestra más grande identifica mejor un efecto fijo

Con un tamaño de muestra suficientemente grande, casi cualquier tamaño de efecto puede parecer estadísticamente significativo. Sin embargo, un tamaño de efecto muy pequeño puede ser inútil en un sentido práctico. Es importante considerar un tamaño de efecto apropiado para el problema en cuestión.

3.3.6 ANOVA

los **pruebas de hipótesis** presentados en las secciones anteriores son buenos para analizar las medias entre dos poblaciones. Pero, ¿y si hay más de dos poblaciones? Considere un ejemplo de prueba del impacto de

nutrición y ejercicio en 60 candidatos de entre 18 y 50 años. Los candidatos se dividen al azar en seis grupos, cada uno asignado con una estrategia de pérdida de peso diferente, y el objetivo es determinar qué estrategia es la más eficaz.

- El grupo 1 solo come comida chatarra.
- El grupo 2 solo come alimentos saludables.
- El grupo 3 come comida chatarra y hace ejercicio cardiovascular día por medio.
- El grupo 4 come alimentos saludables y hace ejercicio cardiovascular día por medio.
- El grupo 5 come comida chatarra y hace ejercicios cardiovasculares y de fuerza cada dos días.
- El grupo 6 come alimentos saludables y hace ejercicios cardiovasculares y de fuerza cada dos días.

Múltiple *t*- Las pruebas podrían aplicarse a cada par de estrategias de pérdida de peso. En este ejemplo, la pérdida de peso del Grupo 1 se compara con la pérdida de peso del Grupo 2, 3, 4, 5 o 6. De manera similar, la pérdida de peso del Grupo 2 se compara con la de los siguientes 4 grupos. Por tanto, un total de 15 *t*- se realizarían pruebas.

Sin embargo, varios *t*- las pruebas pueden no funcionar bien en varias poblaciones por dos razones. Primero, porque el número de *t*- pruebas aumenta a medida que aumenta el número de grupos, el análisis utilizando los múltiples *t*- las pruebas se vuelven cognitivamente más difíciles. En segundo lugar, al realizar un mayor número de análisis, la probabilidad de cometer al menos un error de tipo I en algún lugar del análisis aumenta enormemente.

Análisis de variación (**ANOVA**) está diseñado para abordar estos problemas. ANOVA es una generalización de la prueba de hipótesis de la diferencia de dos medias poblacionales. ANOVA prueba si alguna de las medias de la población difiere de las otras medias de la población. La hipótesis nula de ANOVA es que todas las medias poblacionales son iguales. La hipótesis alternativa es que al menos un par de medias poblacionales no es igual. En otras palabras,

- $H_0: \mu_1 = \mu_2 = \dots = \mu_{norte}$
- $H_{\text{alt}}: \mu_{yo} \neq \mu_j$ por al menos un par de yo, j

Como se vio en la Sección 3.3.2, "Diferencia de medias", se supone que cada población tiene una distribución normal con la misma varianza.

Lo primero que se debe calcular para el ANOVA es el estadístico de prueba. Básicamente, el objetivo es probar si los conglomerados formados por cada población están agrupados de forma más estrecha que la distribución entre todas las poblaciones.

Sea el número total de poblaciones k . El número total de muestras *norte* se divide aleatoriamente en k grupos. El número de muestras en el yo - El grupo se denota como *norte_{yo}*, y la media del grupo es X_{yo} , donde $yo \in [1, k]$. La media de todas las muestras se denota como X_0 .

los **entre grupos media suma de cuadrados**, S_B^2 SEGUNDO es una estimación del **varianza entre grupos**. Eso mide cómo varían las medias de la población con respecto a la gran media, o la media distribuida entre todas las poblaciones. Formalmente, esto se presenta como se muestra en la Ecuación 3-4.

$$S_B^2 = \frac{1}{k-1} \sum_{yo=1}^k norte_{yo} \cdot (X_{yo} - X_0)^2 \quad (3-4)$$

los **dentro del grupo media suma de cuadrados**, S_w^2 es una estimación del **Varianza dentro del grupo**. Cuanti fi ca la dispersión de valores dentro de los grupos. Formalmente, esto se presenta como se muestra en la Ecuación 3-5.

$$S_{W^2} = \frac{1}{n - k} \sum_{j=1}^k \sum_{y_0=1}^{n_j} (X_{jy_0} - \bar{X}_{jy})^2 \quad (3-5)$$

Si S_{W^2} es mucho más grande que S_{B^2} , entonces algunas de las medias de la población son diferentes entre sí.

Los F -el estadístico de prueba se define como la razón de la suma de cuadrados media entre grupos y la suma de cuadrados media dentro del grupo. Formalmente, esto se presenta como se muestra en la Ecuación 3-6.

$$F = \frac{S_B^2}{S_W^2} \quad (3-6)$$

Los F -la estadística de prueba en ANOVA se puede considerar como una medida de cuán diferentes son los medios en relación con la variabilidad dentro de cada grupo. Cuanto mayor sea el observado F -estadística de prueba, mayor es la probabilidad de que las diferencias entre las medias se deban a algo más que al azar. Los F -el estadístico de prueba se usa para probar la hipótesis de que los efectos observados no se deben al azar, es decir, si las medias son significativamente diferentes entre sí.

Considere un ejemplo de que cada cliente que visita un sitio web minorista obtiene una de las dos ofertas promocionales o no obtiene ninguna promoción. El objetivo es ver si hacer las ofertas promocionales marca la diferencia. Se podría utilizar ANOVA y la hipótesis nula es que ninguna promoción marca la diferencia. El código que sigue genera aleatoriamente un total de 500 observaciones de tamaños de compra en tres opciones de oferta diferentes.

```
ofertas <- muestra (c ("oferta1", "oferta2", "nopromo"), tamaño = 500, reemplazo = T)
```

```
# 500 observaciones simuladas de tamaños de compra en las 3 opciones de oferta
shoppingize <- ifelse (offers == "offer1", rnorm (500, mean = 80, sd = 30),
```

```
ifelse (offers == "offer2", rnorm (500, mean = 85, sd = 30),
rnorm (500, media = 40, sd = 30)))
```

```
# crear un marco de datos de opción de oferta y tamaño de compra
offertest <- data.frame (oferta = as.factor (ofertas),
compra_amt = tamaño de compra)
```

El resumen de la **offertest** marco de datos muestra que 170 oferta1, 161 oferta2, y 169 nopromo no se han realizado promociones). También muestra el rango de tamaño de compra (**compra_amt**) para cada una de las tres opciones ofrecidas.

```
# mostrar un resumen de la prueba de oferta donde oferta = "oferta1"
```

```
resumen (offertest [offertest $ oferta == "oferta1",])
```

oferta		purchase_amt
nopromo:	0	Min. : 4.521
oferta1: 1	70	1er Qu.: 58.158
oferta2: 0		Mediana: 76.944
		Media : 81.936
		3er Qu.: 104.959
		Max. : 180.507

```
# mostrar un resumen de la prueba de oferta donde oferta = "oferta2"
```

```
resumen (offertest [offertest $ oferta == "oferta2",])
```

```

oferta      purchase_amt
nopromo: 0   Min.       : 14.04
oferta1: 0    1er Qu.: 69,46
oferta2: 161   Mediana: 90,20
                  Media     : 89,09
                  3er Qu.:107,48
                  Max.      : 154,33

```

mostrar un resumen de offertest donde offer = "nopromo"

```
resumen (offertest [offertest $ oferta == "nopromo",])
```

```

oferta      purchase_amt
nopromo: 169  Min.       : -27,00
oferta1: 0    1er Qu.: 20,22
oferta2: 0    Mediana: 42,44
                  Media     : 40,97
                  3er Qu.: 58,96
                  Max.      : 164,04

```

los aov () La función realiza el ANOVA sobre el tamaño de la compra y las opciones de oferta.

prueba de ajuste ANOVA

```
modelo <- aov (purchase_amt ~ offers, data = offertest)
```

los resumen() La función muestra un resumen del modelo. Los grados de libertad para las ofertas es 2, que corresponde al $k - 1$ en el denominador de la ecuación 3-4. Los grados de libertad para los residuos es 497, que corresponde a la $n_{\text{orte}} - k$ en el denominador de la ecuación 3-5.

resumen (modelo)

	Df	Suma Sq	Valor medio	Sq F	Pr (> F)
ofertas	1	112611	130,6	<2e-16 ***	
Derechos residuales de aov	497	428470	862		
	- - -				

Signif. códigos: 0 ***** 0,001 *** 0,01 ** 0,05 * 0,1 pulg. 1

La salida también incluye el S_2 , ($112,611$), $S_2 - w$, (862), el F - estadístico de prueba (130.6), y el p_{ags} - valor (<2e – 16).

los F -la estadística de prueba es mucho mayor que 1 con un p_{ags} - valor mucho menor que 1. Por lo tanto, la hipótesis nula de que las medias son iguales debe rechazarse.

Sin embargo, el resultado no muestra si oferta1 es diferente de oferta2, que requiere pruebas adicionales. los TukeyHSD () La función implementa la Diferencia Significativa Honesta (HSD) de Tukey en todas las pruebas de pares para la diferencia de medias.

TukeyHSD (modelo)

Comparaciones múltiples de medias de Tukey

95% de nivel de confianza familiar

Ajuste: aov (fórmula = purchase_amt ~ offers, data = offertest)

\$ ofertas	diff	lwr	upr	p adj
oferta1-nopromo	40.961437	33.4638483	48.45903	0.0000000

oferta2-nopromo 48.120286 40.5189446 55.72163 0.0000000 oferta2-oferta1
 7.158849 -0.4315769 14.74928 0.0692895

El resultado incluye *pags*-valores de las comparaciones por pares de las tres opciones ofrecidas. los *pags*-valores para oferta1-nopromo y oferta-nopromo son iguales a 0, menores que el nivel de significancia 0.05. Esto sugiere que tanto oferta1 y oferta2 son signi fi cativamente diferentes de nopromo. UN *pags*- valor de 0.0692895 para oferta2 en contra oferta1 es mayor que el nivel de significancia 0.05. Esto sugiere que oferta2 es *no* signi fi cativamente diferente de oferta 1.

Debido a que solo se ejecutó la influencia de un factor (ofertas), el ANOVA presentado se conoce como ANOVA de una vía. Si el objetivo es analizar dos factores, como las ofertas y el día de la semana, sería un ANOVA de dos factores [16]. Si el objetivo es modelar más de una variable de resultado, se podría utilizar ANOVA multivariante (o MANOVA).

Resumen

R es un paquete popular y un lenguaje de programación para exploración, análisis y visualización de datos. Como introducción a R, este capítulo cubre GUI, E / S de datos, atributos y tipos de datos, y estadísticas descriptivas. Este capítulo también analiza cómo usar R para realizar análisis de datos exploratorios, incluido el descubrimiento de datos sucios, la visualización de una o más variables y la personalización de la visualización para diferentes audiencias. Finalmente, el capítulo presenta algunos métodos estadísticos básicos. El primer método estadístico que se presenta en el capítulo es la prueba de hipótesis. Los estudiantes *t*-prueba y Welch *t*-Las pruebas se incluyen como dos pruebas de hipótesis de ejemplo diseñadas para probar la diferencia de medias. Otros métodos y herramientas estadísticos presentados en este capítulo incluyen intervalos de confianza, prueba de suma de rangos de Wilcoxon, errores de tipo I y II, tamaño del efecto y ANOVA.

Ejercicios

1. ¿Cuántos niveles hace **fdata** contener en el siguiente código R?

```
datos = c(1,2,2,3,1,2,3,3,1,2,3,3,1)
fdata = factor(datos)
```

2. Dos vectores, **v1** y **v2**, se crean con el siguiente código R:

```
v1 <- 1: 5
v2 <- 6: 2
```

¿Cuáles son los resultados de cbind (v1, v2) y rbind (v1, v2)?

3. ¿Qué comando (s) de R usaría para eliminar valores nulos de un conjunto de datos?

4. ¿Qué comando R se puede usar para instalar un paquete R adicional?

5. ¿Qué función R se usa para codificar un vector como categoría?

6. ¿Para qué se usa un diagrama de alfombra en un diagrama de densidad?

7. Un minorista en línea desea estudiar los comportamientos de compra de sus clientes. La figura 3-27 muestra la densidad de tamaño de compra (en dólares). ¿Cuál sería su recomendación para mejorar la trama para detectar más estructuras que de otro modo podrían perderse?

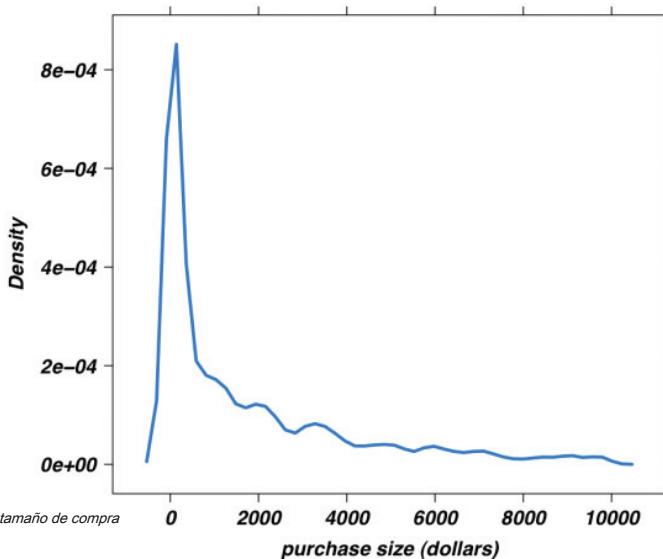


FIGURE 3-27 Parcela de densidad del tamaño de compra

8. ¿En cuántas secciones divide los datos una caja y un bigote? ¿Cuáles son estas secciones?
9. ¿Qué atributos están correlacionados según la Figura 3-18? ¿Cómo describiría sus relaciones?
10. ¿Qué función se puede utilizar para ajustar una línea no lineal a los datos?
11. Si un gráfico de datos está sesgado y todos los datos son positivos, ¿qué técnica matemática se puede utilizar para ayudar a detectar estructuras que de otro modo podrían pasarse por alto?
12. ¿Qué es un error de tipo I? ¿Qué es un error de tipo II? ¿Uno siempre es más serio que el otro? ¿Por qué?
13. Suponga que todos los que visitan un sitio web minorista obtienen una oferta promocional o ninguna promoción. Nosotros queremos ver si hacer una oferta promocional marca la diferencia. ¿Qué método estadístico recomendaría para este análisis?
14. Está analizando dos poblaciones distribuidas normalmente y su hipótesis nula es que la media μ_1 de la primera población es igual a la media μ_2 del segundo. Suponga que el nivel de significancia se establece en 0.05. Si el observado p_{ags} - el valor es 4.33e-05, ¿cuál será su decisión con respecto a la hipótesis nula?

Bibliografía

- [1] El Proyecto R para Computación Estadística, "Licencias R". [En línea]. Disponible: <http://www.r-project.org/Licenses/>. [Consultado el 10 de diciembre de 2013].
- [2] El Proyecto R para Computación Estadística, "La Red Integral de Archivos R". [En línea]. Disponible: <http://cran.r-project.org/>. [Consultado el 10 de diciembre de 2013].

- [3] J. Fox y M. Bouchet-Valat, "The R Commander: una GUI de estadísticas básicas para R", CRAN. [En línea]. Disponible: <http://socserv.mcmaster.ca/jfox/Misc/Rcmdr/>. [Consultado el 11 de diciembre de 2013].
- [4] G. Williams, MV Culp, E. Cox, A. Nolan, D. White, D. Medri y A. Waljee, "Rattle: Interfaz gráfica de usuario para minería de datos en R", CRAN. [En línea]. Disponible: <http://cran.r-project.org/web/packages/rattle/index.html>. [Consultado el 12 de diciembre de 2013].
- [5] RStudio, "RStudio IDE" [en línea]. Disponible: <http://www.rstudio.com/ide/>. [Accedido 11 Diciembre 2013].
- [6] R Grupo de interés especial sobre bases de datos (R-SIG-DB), "DBI: R Database Interface". CRAN [en línea]. Disponible: <http://cran.r-project.org/web/packages/DBI/index.html>. [Consultado el 13 de diciembre de 2013].
- [7] B. Ripley, "RODBC: ODBC Database Access", CRAN. [En línea]. Disponible: <http://cran.r-project.org/web/packages/RODBC/index.html>. [Consultado el 13 de diciembre de 2013].
- [8] SS Stevens, "Sobre la teoría de las escalas de medida", *Ciencias*, vol. 103, no. 2684, pág. 677–680, 1946.
- [9] DC Hoaglin, F. Mosteller y JW Tukey, *Comprendiendo del análisis de datos robusto y exploratorio*, Nueva York: Wiley, 1983.
- [10] FJ Anscombe, "Gráficos en análisis estadístico", *El estadístico estadounidense*, vol. 27, no. 1, págs. 17-21, 1973.
- [11] H. Wickham, "ggplot2", 2013. [En línea]. Disponible: <http://ggplot2.org/>. [Accedido 8 Enero 2014].
- [12] WS Cleveland, *Visualización de datos*, Lafayette, IN: Hobart Press, 1993.
- [13] RA Fisher, "El uso de múltiples medidas en problemas taxonómicos", *Anales de Eugenesia*, vol. 7, no. 2, págs. 179-188, 1936.
- [14] BL Welch, "La generalización del problema de "estudiante "cuando se involucran varias variaciones de población diferentes", *Biometrika*, vol. 34, no. 1–2, págs. 28–35, 1947.
- [15] F. Wilcoxon, "Comparaciones individuales por métodos de clasificación", *Boletín de biometría*, vol. 1, no. 6, págs. 80–83, 1945.
- [dieciséis] JJ Faraway, "Practical Regression and Anova Using R", julio de 2002. [En linea]. Disponible: <http://cran.r-project.org/doc/contrib/Faraway-PRA.pdf>. [Consultado el 22 de enero de 2014].

4

Teoría analítica avanzada andMethods: Clustering

Conceptos clave

Centroide

Agrupación

K-significa

Sin supervisión

Dentro de la suma de cuadrados

A partir de la introducción a R presentada en el Capítulo 3, "Revisión de métodos analíticos de datos básicos mediante R", del Capítulo 4, "Teoría y métodos analíticos avanzados: agrupamiento" hasta el Capítulo 9, "Teoría y métodos analíticos avanzados: análisis de texto", se describen varios métodos analíticos de uso común que pueden considerarse para la planificación del modelo, y Fases de ejecución (Fases 3 y 4) del ciclo de vida del análisis de datos. Este capítulo considera técnicas y algoritmos de agrupación en clústeres.

4.1 Descripción general de la agrupación en clústeres

En general, la agrupación es el uso de **sin supervisión** técnicas para agrupar objetos similares. En el aprendizaje automático, sin supervisión se refiere al problema de encontrar estructuras ocultas dentro de los datos sin etiquetar. Las técnicas de agrupación en clústeres no están supervisadas en el sentido de que el científico de datos no determina, de antemano, las etiquetas que se aplicarán a los clústeres. La estructura de los datos describe los objetos de interés y determina la mejor forma de agruparlos. Por ejemplo, según los ingresos personales de los clientes, es sencillo dividir a los clientes en tres grupos según los valores seleccionados arbitrariamente. Los clientes se pueden dividir en tres grupos de la siguiente manera:

- Gana menos de \$ 10,000
- Gana entre \$ 10,000 y \$ 99,999
- Gana \$ 100,000 o más

En este caso, los niveles de ingresos se eligieron de forma algo subjetiva sobre la base de puntos de delineación fáciles de comunicar. Sin embargo, tales agrupaciones no indican una afinidad natural de los clientes dentro de cada grupo. En otras palabras, no hay ninguna razón inherente para creer que el cliente que gana \$ 90 000 se comportará de manera diferente al cliente que gana \$ 110 000. A medida que se introducen dimensiones adicionales agregando más variables sobre los clientes, la tarea de encontrar agrupaciones significativas se vuelve más compleja. Por ejemplo, suponga que, junto con la variable de ingreso personal, se consideran variables como la edad, los años de educación, el tamaño del hogar y los gastos de compra anuales. ¿Cuáles son las agrupaciones naturales de clientes? Este es el tipo de pregunta que el análisis de agrupamiento puede ayudar a responder.

La agrupación en clústeres es un método que se utiliza a menudo para el análisis exploratorio de los datos. En la agrupación, no se hacen predicciones. Más bien, los métodos de agrupamiento encuentran las similitudes entre los objetos de acuerdo con los atributos del objeto y agrupan los objetos similares en grupos. Las técnicas de agrupación se utilizan en marketing, economía y diversas ramas de la ciencia. Un método de agrupamiento popular es k-means.

4.2 K-medias

Dada una colección de objetos, cada uno con atributos no medibles, **k-means** [1] es una técnica analítica que, para un valor elegido de k, identifica k grupos de objetos en función de la proximidad de los objetos al centro de los k grupos. El centro se determina como el promedio aritmético (media) del vector de atributos n-dimensional de cada grupo. Esta sección describe el algoritmo para determinar los kmedios, así como la mejor forma de aplicar esta técnica a varios casos de uso. La figura 4-1 ilustra tres grupos de objetos con dos atributos. Cada objeto en el conjunto de datos está representado por un pequeño punto codificado por colores al punto grande más cercano, la media del grupo.

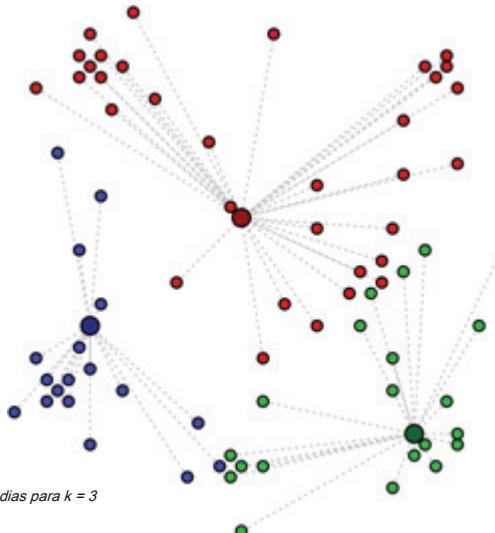


FIGURE 4-1 Posibles grupos de k-medias para $k = 3$

4.2.1 Casos de uso

La agrupación se utiliza a menudo como una introducción a la clasificación. Una vez que se identifican los conglomerados, se pueden aplicar etiquetas a cada conglomerado para clasificar cada grupo en función de sus características. La clasificación se cubre con más detalle en el Capítulo 7, "Teoría y métodos analíticos avanzados: clasificación". La agrupación en clústeres es principalmente una técnica exploratoria para descubrir estructuras ocultas de los datos, posiblemente como preludio de un análisis o procesos de decisión más centrados. Algunas aplicaciones específicas de k-means son procesamiento de imágenes, médico y segmentación de clientes.

Procesamiento de imágenes

El video es un ejemplo de los crecientes volúmenes de datos no estructurados que se recopilan. Dentro de cada cuadro de un video, el análisis de k-medias se puede utilizar para identificar objetos en el video. Para cada cuadro, la tarea consiste en determinar qué píxeles son más similares entre sí. Los atributos de cada píxel pueden incluir brillo, color y ubicación, las coordenadas xey en el marco. Con imágenes de video de seguridad, por ejemplo, se examinan fotogramas sucesivos para identificar cualquier cambio en los grupos. Estos grupos recientemente identificados pueden indicar acceso no autorizado a una instalación.

Médico

Los atributos del paciente, como la edad, la altura, el peso, la presión arterial sistólica y diastólica, el nivel de colesterol y otros atributos, pueden identificar grupos de origen natural. Estos grupos podrían utilizarse para seleccionar a individuos para medidas preventivas específicas o para participar en ensayos clínicos. La agrupación, en general, es útil en biología para la clasificación de plantas y animales, así como en el campo de la genética humana.

Segmentación de clientes

Los grupos de marketing y ventas utilizan k-medias para identificar mejor a los clientes que tienen comportamientos y patrones de gasto similares. Por ejemplo, un proveedor inalámbrico puede considerar los siguientes atributos del cliente: factura mensual, cantidad de mensajes de texto, volumen de datos consumidos, minutos usados durante varios períodos diarios y años como cliente. La compañía inalámbrica podría entonces mirar los clústeres que ocurren naturalmente y considerar tácticas para aumentar las ventas o reducir el número de clientes. **tasa de abandono**, la proporción de clientes que terminan su relación con una empresa en particular.

4.2.2 Descripción general del método

Para ilustrar el método para encontrar k conglomerados de una colección de M objetos con n atributos, se examina el caso bidimensional ($n = 2$). Es mucho más fácil visualizar el método de k-medias en dos dimensiones. Más adelante en el capítulo, el escenario de dos dimensiones se generaliza para manejar cualquier número de atributos.

Debido a que cada objeto en este ejemplo tiene dos atributos, es útil considerar que cada objeto corresponde ir al grano X_{yi}, Y_{yi} , donde $xe y$ denotan los dos atributos $i = 1, 2 \dots M$. Para un grupo dado de m puntos ($m \leq M$), el punto que corresponde a la media del conglomerado se llama **centroide**. En matemáticas, un centroide se refiere a un punto que corresponde al centro de masa de un objeto.

El algoritmo de k-medias para encontrar k conglomerados se puede describir en los siguientes cuatro pasos.

1. Elija el valor de k y las k estimaciones iniciales para los centroides.

En este ejemplo, $k = 3$, y los centroides iniciales están indicados por los puntos sombreados en rojo, verde y azul en la Fig.

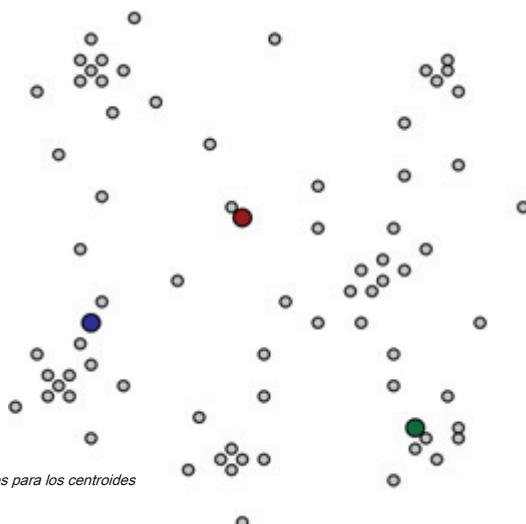


FIGURE 4-2 Puntos de partida iniciales para los centroides

2. Calcule la distancia desde cada punto de datos (X_{yo}, y_{yo}) a cada centroide. Asigne cada punto al centroide más cercano. Esta asociación define los primeros k conglomerados.

En dos dimensiones, la distancia, r_e , entre dos puntos cualesquiera, (X_1, y_1) y (X_2, y_2), en el plano cartesiano se expresa típicamente usando la medida de distancia euclídea proporcionada en la Ecuación 4-1.

(4-1)

En la Figura 4-3,

encontrar el color.

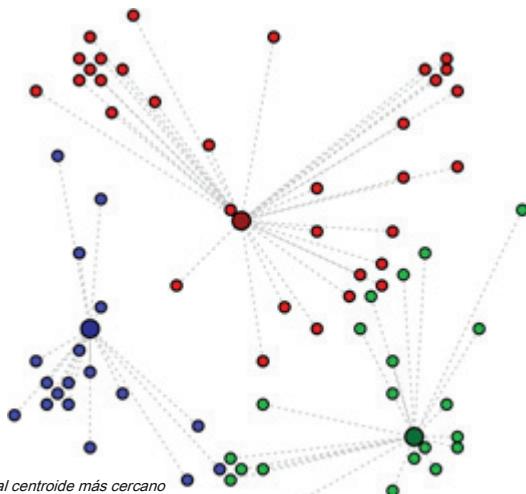


FIGURE 4-3 Los puntos se asignan al centroide más cercano

3. Calcule el centroide, el centro de masa, de cada grupo de fi nido recientemente en el Paso 2.

En la Figura 4-4, los centroides calculados en el Paso 3 son los puntos ligeramente sombreados de los ing color. En dos dimensiones, el centroide (X_c, y_c) de los m puntos en un grupo de k-medias se calcula como sigue en la Ecuación 4-2.

$$(x_c, y_c) = \left(\frac{\sum_{y_o=1}^m x_o}{m}, \frac{\sum_{y_o=1}^m y_o}{m} \right) \quad (4-2)$$

Por lo tanto, (X_c, y_c) es el par ordenado de las medias aritméticas de las coordenadas de los m puntos del grupo. En este paso, se calcula un centroide para cada uno de los k grupos.

4. Repita los pasos 2 y 3 hasta que el algoritmo converja en una respuesta.

a. Asigne cada punto al centroide más cercano calculado en el Paso 3.

segundo. Calcule el centroide de los conglomerados recién definidos.

C. Repita hasta que el algoritmo alcance la respuesta final.

La convergencia se alcanza cuando los centroides calculados no cambian o los centroides y el punto asignado

Siguiente. El último caso puede ocurrir
cuando hay un
el centroide calculado.

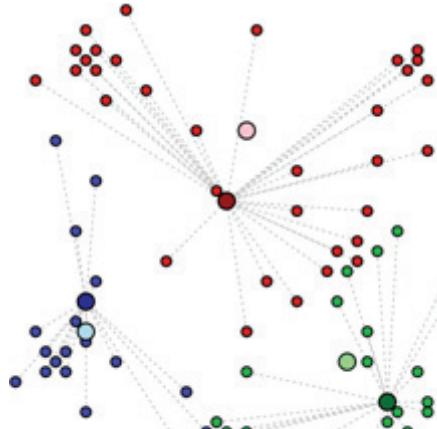


FIGURE 4-4 Calcule el tamaño de cada grupo

Para generalizar el algoritmo anterior en n dimensiones, suponga que hay M objetos, donde cada objeto es descrito por n atributos o valores de propiedad ($pags_{yo1}, pags_{yo2}, \dots, pags_{yo_n}$). Entonces objeto yo es descrito por ($pags_{yo1}, pags_{yo2}, \dots, pags_{yo_n}$) para $yo = 1, 2, \dots, M$. En otras palabras, hay una matriz con M filas correspondientes a los M objetos y n columnas para almacenar los valores de los atributos. Para expandir el proceso anterior para encontrar k conglomerados de dos dimensiones a n dimensiones, las siguientes ecuaciones proporcionan las fórmulas para calcular las distancias y las ubicaciones de los centroides para $n \geq 1$.

Para un punto dado, $pags_{yo}$, a ($pags_{yo1}, pags_{yo2}, \dots, pags_{yo_n}$) y un centroide, q , situado en (q_1, q_2, \dots, q_n), la distancia, re . Entre $pags_{yo}$ y q , se expresa como se muestra en la Ecuación 4-3.

$$d(pags_{yo}, q) = \sum_{j=1}^{norte} (pags_{yo_j} - q_j)^2 \quad (4-3)$$

El centroide, q , de un grupo de m puntos, ($pags_{yo1}, pags_{yo2}, \dots, pags_{yo_m}$) se calcula como se muestra en la Ecuación 4-4.

$$(q_{norte},) = \frac{\sum_{yo=1}^m pags_{yo}}{m} \quad (4-4)$$

4.2.3 Determinación del número de clústeres

Con el algoritmo anterior, se pueden identificar k conglomerados en un conjunto de datos dado, pero ¿qué valor de k se debe seleccionar? El valor de k puede elegirse basándose en una suposición razonable o en algún requisito predefinido. Sin embargo, incluso entonces, sería bueno saber cuánto mejor o peor sería tener k conglomerados versus $k - 1$ o $k + 1$ conglomerados para explicar la estructura de los datos. A continuación, se examina una heurística que utiliza la métrica Dentro de la suma de cuadrados (WSS) para determinar un valor razonablemente óptimo de k . Usando la función de distancia dada en la Ecuación 4-3, WSS se define como se muestra en la Ecuación 4-5.

$$WSS = \sum_{j_0=1}^{METRO} d(pq_{j_0})^2 = \sum_{j_0=1}^{METRO} \sum_{j=1}^{n_{\text{datos}}} q_{j_0 j}^2 \quad (4-5)$$

En otras palabras, WSS es la suma de los cuadrados de las distancias entre cada punto de datos y el centroide más cercano. El término $q_{j_0 j}$ indica el centroide más cercano que está asociado con el j_0 punto. Si los puntos están relativamente cerca de sus respectivos centroides, el WSS es relativamente pequeño. Por lo tanto, si $k + 1$ conglomerados no reducen mucho el valor de WSS en el caso con solo k conglomerados, puede haber pocos beneficios al agregar otro conglomerado.

Uso de R para realizar análisis de K-medias

Para ilustrar cómo usar el WSS para determinar un número apropiado, k , de conglomerados, el siguiente ejemplo usa R para realizar un análisis de k-medias. La tarea es agrupar a 620 estudiantes de último año de secundaria según sus calificaciones en tres áreas temáticas: inglés, matemáticas y ciencias. Las calificaciones se promedian durante su carrera de escuela secundaria y asumen valores de 0 a 100. El siguiente código R establece las bibliotecas R necesarias e importa el archivo CSV que contiene las calificaciones.

```
biblioteca (plyr)
biblioteca (ggplot2)
biblioteca (grupos)
biblioteca (celosía)
biblioteca (gráficos)
biblioteca (cuadricula)
biblioteca (gridExtra)
```

```
# importar las calificaciones de los estudiantes
grade_input = as.data.frame (read.csv ("c:/data/grades_km_input.csv"))
```

El siguiente código R formatea las calificaciones para su procesamiento. El archivo de datos contiene cuatro columnas. La primera columna contiene un número de identificación (ID) del estudiante, y las otras tres columnas corresponden a las calificaciones en las tres materias. Debido a que la identificación del estudiante no se usa en el análisis de agrupamiento, se excluye de la matriz de entrada de k-medias, *kmdata*.

```
kmdata_orig = as.matrix (grade_input [, c ("Estudiante", "Inglés", "Matemáticas", "Ciencias")]) kmdata <- kmdata_orig [, 2:4]
```

kmdatas [1:10,]

	Inglés	Matemáticas	Ciencias
[1,]	99	96	97
[2,]	99	96	97
[3,]	98	97	97
[4,]	95 100		95
[5,]	95	96	96
[6,]	96	97	96
[7,]	100	96	97
[8,]	95	98	98
[9,]	98	96	96
[10,]	99	99	95

Para determinar un valor apropiado para k, se usa el algoritmo de k-medias para identificar grupos para $k = 1, 2, \dots, 15$. Para cada valor de k, se calcula el WSS. Si un clúster adicional proporciona una mejor partición de los puntos de datos, el WSS debería ser notablemente más pequeño que sin el clúster adicional.

El siguiente código R recorre varios análisis de k-medias para el número de centroides, k , variando de 1 a 15. Para cada k, la opción `nstart = 25` especifica que el algoritmo de k-medias se repetirá 25 veces, cada una comenzando con k centroides iniciales aleatorios. El valor correspondiente de WSS para cada análisis de k-media se almacena en el `wss` vector.

```
wss <- numérico (15)
for (k en 1:15) wss [k] <- sum (kmeans (kmdatas, centers = k, nstart = 25) $ withinss)
```

Utilizando la función de gráfico R básica, cada WSS se grafica contra el número respectivo de centroides, 1 a 15. Este gráfico se muestra en la Figura 4-5.

```
trama (1:15, wss, t
Cuadricula")
```

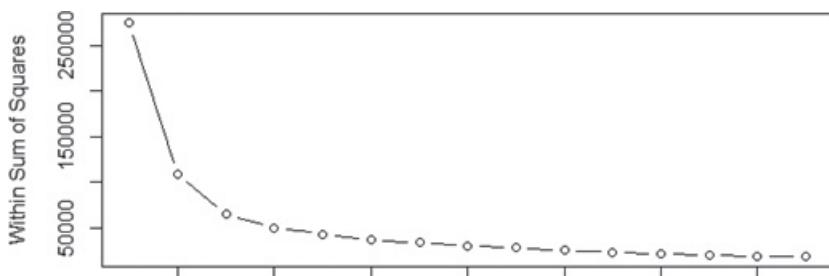


FIGURE 4-5 WSS de los datos de calificaciones del estudiante

Como puede verse, el WSS se reduce en gran medida cuando k aumenta de uno a dos. Otra reducción sustancial en WSS ocurre en k = 3. Sin embargo, la mejora en WSS es bastante lineal para k > 3. Por lo tanto, el análisis de k-medias se realizará para k = 3. El proceso de identificar el valor apropiado de k es referido como encontrar el "codo" de la curva WSS.

```
km = kmeans (kmdata, 3, nstart = 25) km
```

Agrupación de K-medias con 3 agrupaciones de tamaños 158, 218, 244

Clúster significa:

Inglés	Ciencias Matemáticas
1 97.21519	93.37342
94.86076	
2 73.22018	64.62844
	65.84862
3 85.84426	79.68033
	81.50820

Vector de agrupación:

Suma de cuadrados dentro del grupo por grupo: [1] 6692.589
34806.339 22984.131 (between SS / total SS = 76.5%)

Componentes disponibles:

```
[1] "grupo"           "centros" "totss"           "sin"      "tot.withinss"
[6] "entre" "tamaño" "iter"      "ifault"
```

El contenido mostrado de la variable *km* incluya lo siguiente:

- La ubicación del clúster significa
- Un vector de agrupamiento que define la pertenencia de cada estudiante a un grupo correspondiente 1, 2 o 3.
- EWSS de cada clúster
- Una lista de todos los componentes k-means disponibles

El lector puede encontrar detalles sobre estos componentes y utilizar k-medias en R empleando la función de ayuda. El lector puede haberse preguntado si los resultados de k-medias almacenados en *km* son equivalentes a los resultados de WSS obtenidos anteriormente al generar el gráfico de la Figura 4-5. La siguiente verificación verifica que los resultados sean realmente equivalentes.

```
c (wss [3], suma (km $ sin))
```

```
[1] 64483.06 64483.06
```

Al determinar el valor de k, el científico de datos debe visualizar los datos y los grupos asignados. En el siguiente código, el ggplot2 El paquete se utiliza para visualizar los grupos y centroides de estudiantes identificados.

```
# preparar los datos de los estudiantes y los resultados de la agrupación para graficar
df = as.data.frame (kmdatas_orig [, 2: 4])
df $ cluster = factor (km $ cluster)
centros = as.data.frame (km $ centros)

g1 = ggplot (data = df, aes (x = English, y = Math, color = cluster)) +
  geom_point () + tema (leyenda.posición = "derecha") + geom_point (datos =
  centros,
    aes (x = inglés, y = matemáticas, color = as.factor (c (1,2,3))),
    tamaño = 10, alpha = .3, show_guide = FALSE)

g2 = ggplot (data = df, aes (x = inglés, y = ciencia, color = cluster)) +
  geom_point () +
  geom_point (datos = centros,
    aes (x = inglés, y = ciencia, color = as.factor (c (1,2,3))),
    tamaño = 10, alpha = .3, show_guide = FALSE)

g3 = ggplot (data = df, aes (x = Math, y = Science, color = cluster)) +
  geom_point () +
  geom_point (datos = centros,
    aes (x = Matemáticas, y = Ciencias, color = as.factor (c (1,2,3))),
    tamaño = 10, alpha = .3, show_guide = FALSE)

tmp = ggplot_gtable (ggplot_build (g1))
```

```
grid.arrange (organizarGrob (g1 + theme (legend.position = "none"),
                            g2 + tema (legend.position = "none"),
                            g3 + tema (legend.position = "none"),
                            main = "Análisis de conglomerados de estudiantes de secundaria", ncol = 1))
```

Los gráficos resultantes se muestran en la Figura 4-6. Los círculos grandes representan la ubicación de los medios de agrupación proporcionados anteriormente en la pantalla del *km* contenido. Los pequeños puntos representan a los estudiantes correspondientes al grupo apropiado por color asignado: rojo, azul o verde. En general, las gráficas indican los tres grupos de estudiantes: los mejores estudiantes académicos (rojo), los estudiantes con desafíos académicos (verde) y

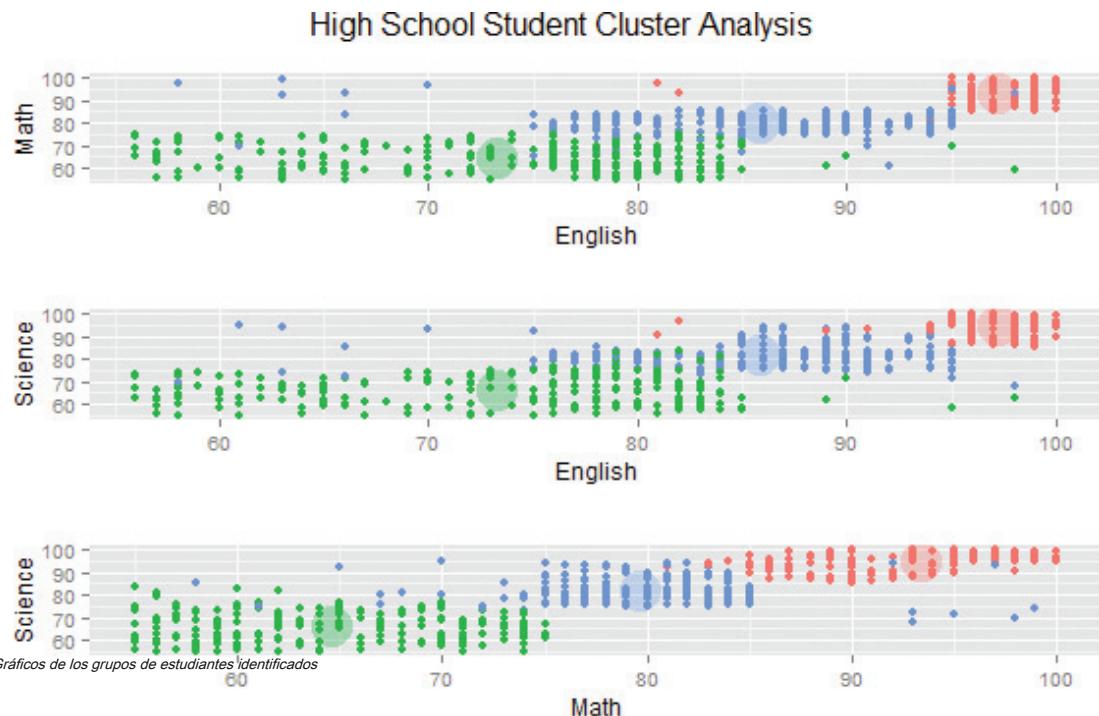


FIGURE 4-6 Gráficos de los grupos de estudiantes identificados

La asignación de etiquetas a los grupos identificados es útil para comunicar los resultados de un análisis. En un contexto de marketing, es común etiquetar a un grupo de clientes como compradores frecuentes o grandes consumidores. Dichas designaciones son especialmente útiles cuando se comunican los resultados de la agrupación en clústeres a los usuarios comerciales o ejecutivos. Es mejor describir el plan de marketing para los grandes gastadores que el Cluster # 1.

4.2.4 Diagnóstico

La heurística que utiliza WSS puede proporcionar al menos varios valores k posibles a considerar. Cuando el número de atributos es relativamente pequeño, un enfoque común para refinar aún más la elección de k es trazar los datos para determinar qué tan distintos son los conglomerados identificados entre sí. En general, se deben considerar las siguientes preguntas.

- ¿Están los racimos bien separados unos de otros?
- ¿Alguno de los grupos tiene solo unos pocos puntos?
- ¿Alguno de los centroides parece estar demasiado cerca el uno del otro?

En el primer caso, lo ideal sería que el gráfico se pareciera al que se muestra en la figura 4-7, cuando $n = 2$. Los conglomerados están bien definidos, con un espacio considerable entre los cuatro conglomerados identificados. Sin embargo, en otros casos, como en la Figura 4-8, el

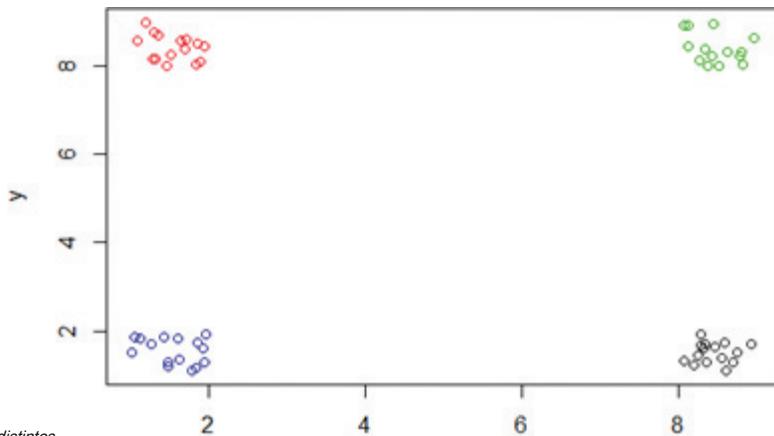


FIGURE 4-7 Ejemplo de clústeres distintos

En tales casos, es importante aplicar algún juicio sobre si se producirá algo diferente al usar más grupos. Por ejemplo, la Figura 4-9 usa seis grupos para describir el mismo conjunto de datos que se usa en la Figura 4-8. Si el uso de más grupos no distingue mejor los grupos, es casi seguro que sea mejor optar por menos grupos.

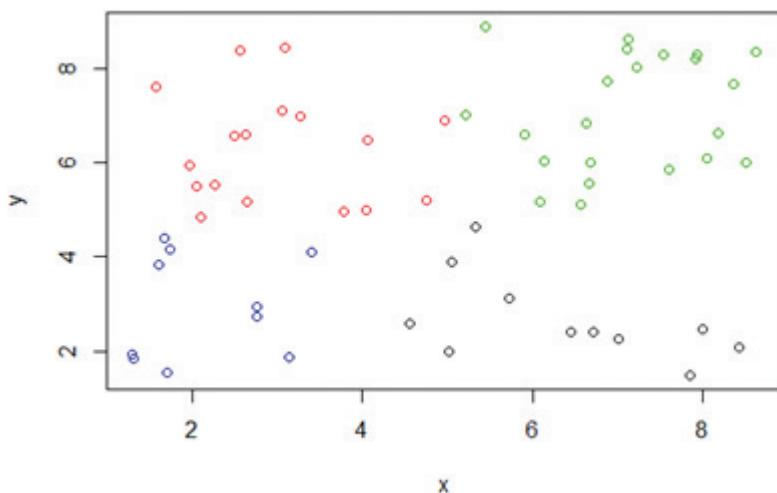


FIGURE 4-8 Ejemplo de los

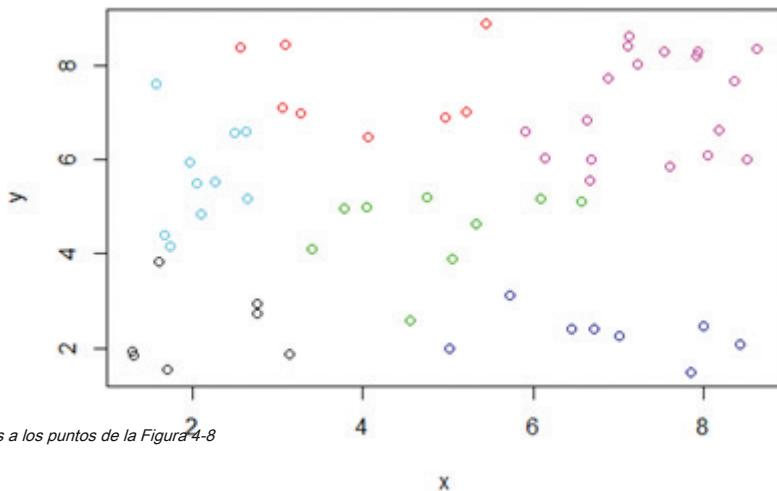


FIGURE 4-9 Seis grupos aplicados a los puntos de la Figura 4-8

4.2.5 Razones para elegir y precauciones

K-means es un método sencillo y directo para definir clústeres. Una vez que se identifican los grupos y sus centroides asociados, es fácil asignar nuevos objetos (por ejemplo, nuevos clientes) a un grupo en función de la distancia del objeto desde el centroide más cercano. Debido a que el método no está supervisado, el uso de k-medias ayuda a eliminar la subjetividad del análisis.

Aunque k-means se considera un método no supervisado, todavía hay varias decisiones que el profesional debe tomar:

- ¿Qué atributos de objeto deben incluirse en el análisis?
- ¿Qué unidad de medida (por ejemplo, millas o kilómetros) debe usarse para cada atributo?
- ¿Es necesario cambiar la escala de los atributos para que un atributo no tenga un efecto desproporcionado en los resultados?

- ¿Qué otras consideraciones pueden aplicarse?

Atributos de objeto

Con respecto a qué atributos del objeto (por ejemplo, edad e ingresos) usar en el análisis, es importante comprender qué atributos se conocerán en el momento en que se asigne un nuevo objeto a un clúster. Por ejemplo, la información sobre la satisfacción de los clientes existentes o la frecuencia de compra puede estar disponible, pero tal información puede no estar disponible para los clientes potenciales.

El científico de datos puede elegir entre una docena o más de atributos para usar en el análisis de agrupamiento. Siempre que sea posible y según los datos, es mejor reducir el número de atributos en la medida de lo posible. Demasiados atributos pueden minimizar el impacto de las variables más importantes. Además, el uso de varios atributos similares puede otorgar demasiada importancia a un tipo de atributo. Por ejemplo, si se incluyen cinco atributos relacionados con la riqueza personal en un análisis de agrupamiento, los atributos de riqueza dominan el análisis y posiblemente enmascaran la importancia de otros atributos, como la edad.

Cuando se trata del problema de demasiados atributos, un enfoque útil es identificar cualquier atributo altamente correlacionado y usar solo uno o dos de los atributos correlacionados en el análisis de agrupamiento. Como se ilustra en la Figura 4-10, una matriz de diagramas de dispersión, como se presentó en el Capítulo 3, es una herramienta útil para visualizar las relaciones por pares entre los atributos.

Se observa que la relación más fuerte se da entre **Atributo3** y **Atributo 7**. Si se conoce el valor de uno de estos dos atributos, parece que el valor del otro atributo se conoce con casi certeza. También se identifican otras relaciones lineales en la gráfica. Por ejemplo, considere la trama de **Atributo2** en contra **Atributo 3**. Si el valor de **Atributo2** se conoce, todavía hay una amplia gama de valores posibles para **Atributo 3**. Por lo tanto, se debe prestar más atención antes de eliminar uno de estos atributos del análisis de agrupamiento.

Otra opción para reducir el número de atributos es combinar varios atributos en una sola medida. Por ejemplo, en lugar de utilizar dos variables de atributo, una para Deuda y otra para Activos, se podría utilizar una relación Deuda a Activo. Esta opción también aborda el problema cuando la magnitud de un atributo no es de interés real, pero la magnitud relativa es una medida más importante.

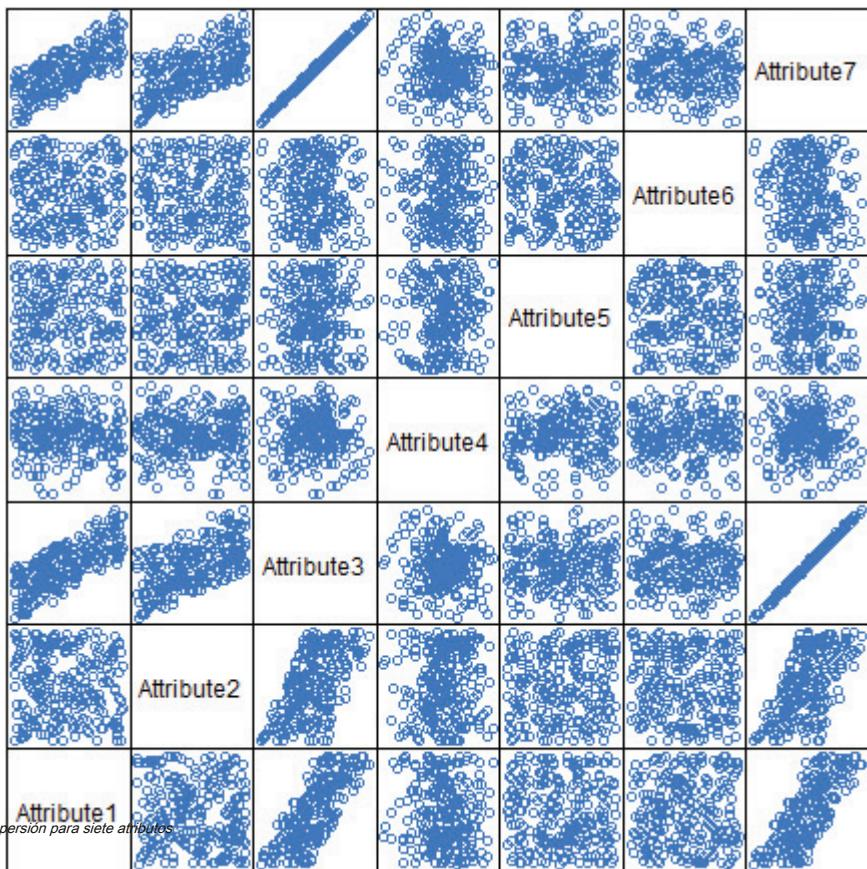


FIGURE 4-10 Matriz de gráficos de dispersión para siete atributos.

Unidades de medida

Desde una perspectiva computacional, el algoritmo de k-medias es algo indiferente a las unidades de medida para un atributo dado (por ejemplo, metros o centímetros para la altura de un paciente). Sin embargo, el algoritmo identificará diferentes grupos dependiendo de la elección de las unidades de medida. Por ejemplo, suponga que se utiliza k-means para clú

4-11

ilustra los dos clu

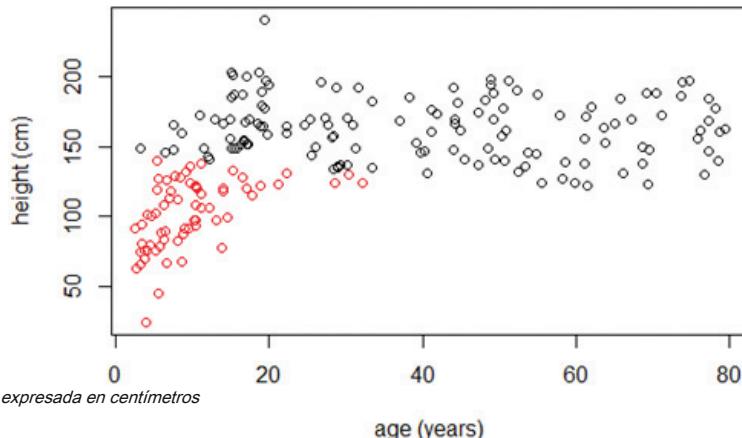


FIGURE 4-11 Grupos con altura expresada en centímetros

Pero si la altura se cambiara de centímetros a metros dividiéndola por 100, los grupos resultantes serían ligeramente diferentes, como se ilustra en la Figura 4-12.

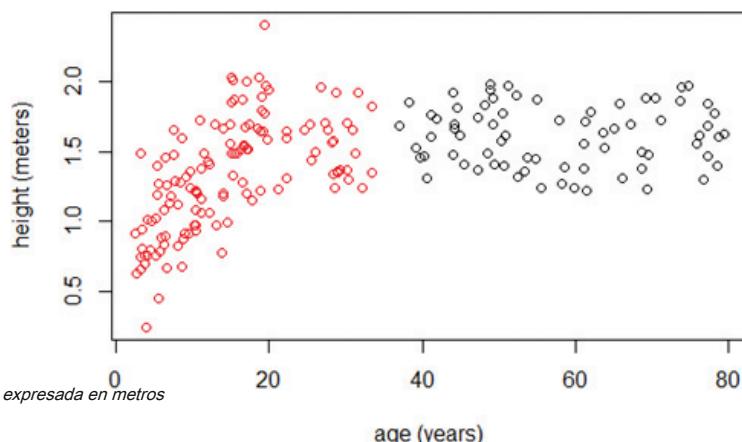


FIGURE 4-12 Grupos con altura expresada en metros

Cuando la altura se expresa en metros, la magnitud de las edades domina el cálculo de la distancia entre dos puntos. El atributo de altura proporciona sólo la medida en que el cuadrado entre la diferencia de la altura máxima y la altura mínima o $(2.0 - 0)^2 = 4$ al radicando, el número debajo del símbolo de raíz cuadrada en la fórmula de distancia dada en la Ecuación 4-3. La edad puede contribuir tanto como $(80 - 0)^2 = 6.400$ al radicando al medir la distancia.

Reescalado

Los atributos que se expresan en dólares son comunes en los análisis de conglomerados y pueden diferir en magnitud de los otros atributos. Por ejemplo, si el ingreso personal se expresa en dólares y la edad en años, el atributo de ingreso, que a menudo excede los \$ 10,000, puede dominar fácilmente el cálculo de la distancia con edades típicamente menores de 100 años.

Aunque se podrían hacer algunos ajustes expresando el ingreso en miles de dólares (por ejemplo, 10 por \$ 10,000), un método más sencillo es dividir cada atributo por la desviación estándar del atributo. Cada uno de los atributos resultantes tendrá una desviación estándar igual a 1 y no tendrá unidades. Volviendo al ejemplo de la edad y la altura, las desviaciones estándar son 23,1 años y 36,4 cm, respectivamente. Dividir cada valor de atributo por la desviación estándar apropiada y realizar el rendimiento del análisis de k-medias

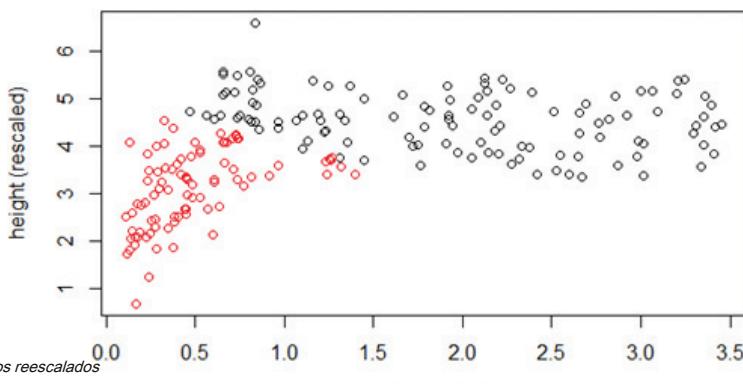


FIGURE 4-13 Clústeres con atributos reescalados

Con los atributos reescalados para edad y altura, los límites de los conglomerados resultantes ahora se encuentran en algún lugar entre los dos análisis de conglomerados anteriores. Tal ocurrencia no es sorprendente en base a las magnitudes de los atributos de los intentos de agrupamiento anteriores. Algunos profesionales también restan las medias de los atributos para centrar los atributos alrededor de cero. Sin embargo, este paso es innecesario porque la fórmula de distancia solo es sensible a la escala del atributo, no a su ubicación.

En muchos análisis estadísticos, es común transformar datos típicamente sesgados, como los ingresos, con colas largas tomando el logaritmo de los datos. Dicha transformación también se puede aplicar en k-medias, pero los datos El científico debe estar consciente del efecto que tendrá esta transformación. Por ejemplo, si *Iniciar sesión*₁₀ del ingreso expresado en dólares, el practicante está esencialmente afirmando que, desde una perspectiva de agrupamiento, \$ 1,000 es tan cercano a \$ 10,000 como \$ 10,000 a \$ 100,000 (*porque* $\log_{10} 1.000 = 3$, *Iniciar sesión*₁₀ 10,000 = 4, y *registro*₁₀ 100,000 = 5). En muchos casos, la asimetría de los datos puede ser la razón para realizar el análisis de agrupamiento en primer lugar.

consideraciones adicionales

El algoritmo de k-medias es sensible a las posiciones iniciales del centroide inicial. Por lo tanto, es importante volver a ejecutar el análisis de k-medias varias veces para un valor particular de k para asegurar que los resultados del conglomerado proporcionen el WSS mínimo general. Como se vio anteriormente, esta tarea se logra en R usando el *nstart* opción en el *kmeans ()* Llamada de función.

Este capítulo presentó el uso de la función de distancia euclídea para asignar los puntos a los centroides más cercanos. Otras posibles opciones de función incluyen la similitud de coseno y las funciones de distancia de Manhattan. La función de similitud de coseno se elige a menudo para comparar dos documentos en función de la frecuencia de cada palabra que aparece en cada uno de los documentos [2]. Por dos puntos, *pags* y *q*, a (*pags*₁, *pags*₂, ... *pags*_n) y (*q*₁, *q*₂, ... *q*_n), respectivamente, la distancia de Manhattan, *re*. Entre *pags* y *q* se expresa como se muestra en la Ecuación 4-6.

$$re(p, q) = \sum_{j=1}^{norte} |pags_j - q_j| \quad (4-6)$$

La función de distancia de Manhattan es análoga a la distancia recorrida por un automóvil en una ciudad, donde las calles se trazan en una cuadrícula rectangular (como las manzanas de una ciudad). En la distancia euclídea, la medición se realiza en línea recta. Usando la ecuación 4-6, la distancia de (1, 1) a (4, 5) sería $|1 - 4| + |1 - 5| = 7$. Desde la perspectiva de la optimización, si es necesario utilizar la distancia Manhattan para un análisis de agrupamiento, la mediana es una mejor opción para el centroide que el uso de la media [2].

La agrupación de K-medias es aplicable a objetos que pueden describirse mediante atributos que son numéricos con una medida de distancia significativa. Del Capítulo 3, ciertamente se pueden usar los tipos de atributos de intervalo y razón. Sin embargo, k-means no maneja bien las variables categóricas. Por ejemplo, suponga que se va a realizar un análisis de agrupamiento en las ventas de automóviles nuevos. Entre otros atributos, como el precio de venta, se considera importante el color del automóvil. Aunque se podrían asignar valores numéricos al color, como rojo = 1, amarillo = 2 y verde = 3, no es útil considerar que el amarillo es tan cercano al rojo como el amarillo al verde desde una perspectiva de agrupamiento. En tales casos, puede ser necesario utilizar una metodología de agrupación en clústeres alternativa. Estos métodos se describen en la siguiente sección.

4.3 Algoritmos adicionales

El método de agrupamiento de k-medias se aplica fácilmente a datos numéricos donde el concepto de distancia se puede aplicar naturalmente. Sin embargo, puede ser necesario o deseable utilizar un algoritmo de agrupamiento alternativo. Como se discutió al final de la sección anterior, k-means no maneja datos categóricos. En tales casos, k-modes [3] es un método comúnmente usado para agrupar datos categóricos basado en el número de diferencias en los componentes respectivos de los atributos. Por ejemplo, si cada objeto tiene cuatro atributos, la distancia de (a, b, e, d) a (d, d, d, d) es 3. En R, la función *kmode ()* se implementa en el *klaR* paquete.

Debido a que k-means y k-modes dividen todo el conjunto de datos en grupos distintos, ambos enfoques se consideran métodos de partición. Un tercer método de partición se conoce como Partitioning Around Medoids (PAM) [4]. En general, un medoide es un objeto representativo en un conjunto de objetos. En agrupamiento, los **medoides** son los objetos de cada grupo que minimizan la suma de las distancias desde el medoide a los otros objetos del grupo. La ventaja de utilizar PAM es que el "centro" de cada grupo es un objeto real en el conjunto de datos. PAM se implementa en R por el pam () función incluida en el racimo Paquete R. los fpc

El paquete R incluye una función pamk (), que usa el pam () función para encontrar el valor óptimo de k.

Otros métodos de agrupamiento incluyen el agrupamiento de aglomeraciones jerárquicas y los métodos de agrupamiento de densidades. En la agrupación jerárquica, cada objeto se coloca inicialmente en su propio grupo. A continuación, los grupos se combinan con el grupo más similar. Este proceso se repite hasta que existe un grupo, que incluye todos los objetos. El r estadísticas el paquete incluye el hclust () función para realizar agrupaciones jerárquicas. En los métodos de agrupamiento basados en la indensidad, los grupos se identifican por la concentración de puntos. Los fpc El paquete R incluye una función, dbscan (), para realizar análisis de agrupación basados en densidad. La agrupación basada en densidad puede resultar útil para identificar agrupaciones de forma irregular.

Resumen

El análisis de agrupación agrupa objetos similares en función de los atributos de los objetos. La agrupación en clústeres se aplica en áreas como marketing, economía, biología y medicina. Este capítulo presentó una explicación detallada del algoritmo de k-medias y su implementación en R. Para usar k-medias correctamente, es importante hacer lo siguiente:

- Escale correctamente los valores de los atributos para evitar que ciertos atributos dominen los otros atributos.
- Asegúrese de que el concepto de distancia entre los valores asignados dentro de un atributo sea significativo.
- Elija el número de conglomerados, k, de modo que la suma de las distancias dentro de la suma de cuadrados (WSS) se minimice razonablemente. Un lote como el del ejemplo de la Figura 4-5 puede ser útil a este respecto.

Si k-means no parece ser una técnica de agrupamiento apropiada para un conjunto de datos dado, entonces se deben considerar técnicas alternativas como k-mode o PAM.

Una vez identificados los conglomerados, a menudo es útil etiquetarlos de alguna manera descriptiva. Especialmente cuando se trata de la alta dirección, estas etiquetas son útiles para comunicar fácilmente los resultados del análisis de agrupamiento. En la agrupación, las etiquetas no se asignan previamente a cada objeto. Las etiquetas se asignan subjetivamente después de que se hayan identificado los grupos. El capítulo 7 considera varios métodos para realizar la clasificación de objetos con etiquetas predeterminadas. La agrupación en clústeres se puede utilizar con otras técnicas analíticas, como la regresión. La regresión lineal y la regresión logística se tratan en el Capítulo 6, "Teoría y métodos analíticos avanzados: regresión".

Ejercicios

1. Utilizando el ejemplo de agrupamiento por edad y altura de la sección 4.2.5, ilustre algebraicamente el impacto en la distancia medida cuando la altura se expresa en metros en lugar de centímetros. Explique por qué se producirán diferentes grupos según la elección de unidades para la altura del paciente.
2. Compare y contraste cinco algoritmos de agrupamiento, asignados por el instructor o seleccionados por el estudiante.

3. Utilizando la ruspini conjunto de datos proporcionado con el racimo paquete en R, realice un análisis de k-medias. Documente los hallazgos y justifique la elección de k. Sugerencia: use datos (ruspini) para cargar el conjunto de datos en Rworkspace.

Bibliografía

- [1] J. MacQueen, "Algunos métodos de clasificación y análisis de observaciones multivariadas", en *Actas del Quinto Simposio de Berkeley sobre Estadística Matemática y Probabilidad*, Berkeley, California, 1967.
- [2] P.-N. Tan, V. Kumar y M. Steinbach, *Introducción a DataMining*, Upper Saddle River, Nueva Jersey: Personia, 2013.
- [3] Z. Huang, "Un algoritmo de agrupamiento rápido para agrupar conjuntos de datos categóricos muy grandes en minería de datos", 1997. [En línea]. Disponible: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.134.83&rep=rep1&type=pdf>. [Consultado el 13 de marzo de 2014].
- [4] L. Kaufman y PJ Rousseeuw, "Partitioning Around Medoids (Program PAM)", en *Encontrar grupos en Datos: Introducción al análisis de conglomerados*, Hoboken, Nueva Jersey, John Wiley & Sons, Inc, 2008, pág. 68-125, Capítulo 2.

5

Avanzado Analítico Teoría y métodos: Reglas de asociación

Conceptos clave

reglas de asociación

Algoritmo a priori

Apoyo

Confianza

Ascensor

Influencia

Este capítulo analiza un método de aprendizaje no supervisado llamado reglas de asociación. Este es un método descriptivo, no predictivo, que se usa a menudo para descubrir relaciones interesantes ocultas en un gran conjunto de datos. Las relaciones reveladas se pueden representar como reglas o conjuntos de elementos frecuentes. Las reglas de asociación se usan comúnmente para minar transacciones en bases de datos.

Aquí hay algunas preguntas posibles que las reglas de asociación pueden responder:

- ¿Qué productos suelen comprarse juntos?
- De aquellos clientes que son similares a esta persona, ¿qué productos suelen comprar?
- De aquellos clientes que han comprado este producto, ¿qué otros productos similares tienden a ver o comprar?

5.1 Resumen

La figura 5-1 muestra la lógica general detrás de las reglas de asociación. Dada una gran colección de transacciones (representadas como tres pilas de recibos en la figura), en las que cada transacción consta de uno o más artículos, las reglas de asociación revisan los artículos que se compran para ver qué artículos se compran juntos con frecuencia y descubrir una lista de reglas que describen el comportamiento de compra. El objetivo de las reglas de asociación es descubrir relaciones interesantes entre los elementos. (La relación ocurre con demasiada frecuencia para ser aleatoria y es significativa desde un

esting depende bot



FIGURE 5-1 La lógica general detrás de las reglas de asociación

Cada una de las reglas descubiertas tiene la forma $X \rightarrow Y$, lo que significa que cuando se observa el elemento X, el elemento Y es también observado. En este caso, el lado izquierdo (LHS) de la regla es X y el lado derecho (RHS) de la regla es Y.

Usando reglas de asociación, se pueden descubrir patrones a partir de los datos que permiten que los algoritmos de reglas de asociación revelen reglas de compras de productos relacionados. Las reglas descubiertas se enumeran en el lado derecho de

Figura 5-1. Las primeras tres reglas sugieren que cuando se compra cereal, el 90% del tiempo también se compra leche. Cuando se compra pan, el 40% del tiempo también se compra leche. Cuando se compra leche, el 23% del tiempo también se compra cereal.

En el ejemplo de una tienda minorista, las reglas de asociación se utilizan en transacciones que constan de uno o más artículos. De hecho, debido a su popularidad en las transacciones de clientes de minería, las reglas de asociación a veces se denominan *análisis de la canasta de mercado*. Cada transacción puede verse como la cesta de la compra de un cliente que contiene uno o más artículos. Esto también se conoce como conjunto de elementos. El término *itemset*

se refiere a una colección de elementos o entidades individuales que contienen algún tipo de relación. Esto podría ser un conjunto de artículos minoristas comprados juntos en una transacción, un conjunto de hipervínculos en los que un usuario hace clic en una sola sesión o un conjunto de tareas realizadas en un día. Un conjunto de elementos que contiene k elementos se llama un *k-itemset*.

Este capítulo utiliza llaves como { artículo 1, artículo 2, ..., artículo k } para denotar un k -itemset. El cálculo de las reglas de asociación se basa normalmente en conjuntos de elementos.

La investigación de las reglas de asociación comenzó ya en la década de 1960. Las primeras investigaciones de Hájek et al. [1] introdujeron muchos de los conceptos y enfoques clave del aprendizaje de reglas de asociación, pero se centró en la representación matemática en lugar del algoritmo. El marco del aprendizaje de reglas de asociación fue introducido en la comunidad de bases de datos por Agrawal et al. [2] a principios de la década de 1990 por descubrir regularidades entre productos en una gran base de datos de transacciones de clientes registradas por sistemas de punto de venta en los supermercados. En años posteriores, se expandió a contextos web, como patrones de recorrido de ruta de minería [3] y patrones de uso [4] para facilitar la organización de páginas web.

Este capítulo elige Apriori como el enfoque principal de la discusión de las reglas de asociación. Apriori [5] es uno de los primeros y más fundamentales algoritmos para generar reglas de asociación. Fue pionera en el uso de soporte para podar los conjuntos de elementos y controlar el crecimiento exponencial de los conjuntos de elementos candidatos. Los conjuntos de elementos candidatos más cortos, que se sabe que son conjuntos de elementos frecuentes, se combinan y podan para generar conjuntos de elementos frecuentes más largos. Este enfoque elimina la necesidad de enumerar todos los conjuntos de elementos posibles dentro del algoritmo, ya que el número de todos los conjuntos de elementos posibles puede llegar a ser exponencialmente grande.

Un componente importante de Apriori es el soporte. Dado un conjunto de elementos L , la *apoyo* [2] de L es el porcentaje de transacciones que contienen L . Por ejemplo, si el 80% de todas las transacciones contienen itemset {pan de molde}, luego el apoyo de { pan de molde} es 0.8. Del mismo modo, si el 60% de todas las transacciones contienen itemset {pan con mantequilla}, luego el apoyo de { pan con mantequilla} es 0.6.

UN *conjunto de elementos frecuentes* tiene elementos que aparecen juntos con bastante frecuencia. El término "bastante a menudo" se define formalmente con un *apoyo mínimo* criterio. Si el soporte mínimo se establece en 0.5, cualquier conjunto de elementos puede considerarse un conjunto de elementos frecuente si al menos el 50% de las transacciones contienen este conjunto de elementos. En otras palabras, el soporte de un conjunto de elementos frecuentes debe ser mayor o igual al soporte mínimo. Para el ejemplo anterior, ambos { pan de molde} y { pan con mantequilla} se consideran conjuntos de elementos frecuentes en el soporte mínimo 0.5. Si el soporte mínimo es 0.7, solo { pan de molde} se considera un conjunto de elementos frecuente.

Si un conjunto de elementos se considera frecuente, cualquier subconjunto del conjunto de elementos frecuentes también debe ser frecuente. Esto se conoce como el *Propiedad Apriori* (o *propiedad de cierre hacia abajo*). Por ejemplo, si el 60% de las transacciones contienen { pan, mermelada}, entonces al menos el 60% de todas las transacciones contendrán { pan de molde} o { mermelada}. En otras palabras, cuando el apoyo de { pan, mermelada} es 0.6, el soporte de { pan de molde} o { mermelada} es al menos 0.6. La figura 5-2 ilustra cómo funciona la propiedad Apriori. Si itemset { B, C, D} es frecuente, entonces todos los subconjuntos de este conjunto de elementos, sombreados, también deben ser conjuntos de elementos frecuentes. La propiedad Apriori proporciona la base para el algoritmo Apriori.

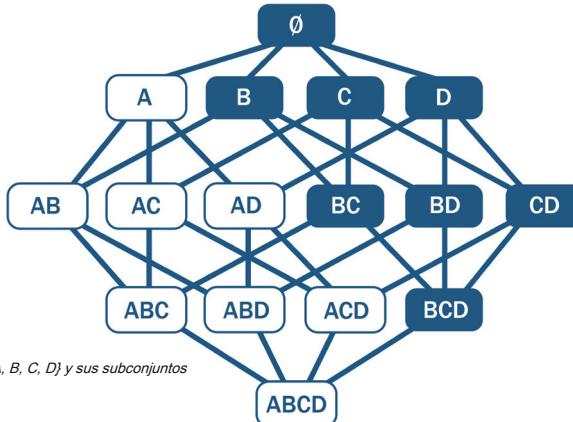


FIGURE 5.2 Conjunto de elementos $\{A, B, C, D\}$ y sus subconjuntos

5.2 Algoritmo a priori

El algoritmo Apriori adopta un enfoque iterativo de abajo hacia arriba para descubrir los conjuntos de elementos frecuentes determinando primero todos los elementos posibles (o conjuntos de 1 elemento, por ejemplo { pan}, {huevos}, {leche}, ...) Y luego identificando cuáles de ellos son frecuentes.

Suponiendo que el umbral de soporte mínimo (o el criterio de soporte mínimo) se establece en 0,5, el algoritmo identifica y retiene los conjuntos de elementos que aparecen en al menos el 50% de todas las transacciones y descarta (o "elimina") los conjuntos de elementos que tienen un soporte menor de 0,5 o aparecen en menos del 50% de las transacciones. La palabra *ciruela pasa* se utiliza como en jardinería, donde se recortan las ramas no deseadas de un arbusto.

En la siguiente iteración del algoritmo Apriori, los conjuntos de 1 elementos frecuentes identificados se emparejan en conjuntos de 2 elementos (por ejemplo, { pan, huevos}, {pan, leche}, {huevos, leche}, ...) Y nuevamente evaluados para identificar los conjuntos de 2 elementos frecuentes entre ellos.

En cada iteración, el algoritmo comprueba si se puede cumplir el criterio de apoyo; Si puede, el algoritmo aumenta el conjunto de elementos, repitiendo el proceso hasta que se quede sin soporte o hasta que los conjuntos de elementos alcancen un valor predefinido.

longitud. A continuación se proporciona el algoritmo Apriori [5]. Deje variable C ser el conjunto de candidatos k - conjuntos de elementos y variables L_k ser el conjunto de k - conjuntos de elementos que satisfacen el soporte mínimo. Dada una base de datos de transacciones RE , un umbral de soporte mínimo δ y un parámetro opcional *norte* indicando la longitud máxima que puede alcanzar un conjunto de elementos,

Apriori calcula iterativamente conjuntos de elementos frecuentes L_{k+1} . Residencia en L_k

- 1 A priori (re , δ , *norte*)
- 2 $k \leftarrow 1$
- 3 $L_k \leftarrow \{ \text{Conjuntos de 1 elemento que satisfacen el soporte mínimo } \delta \}$
- 4 mientras $L_k \neq \emptyset$
- 5 Si *norte* v ($\exists norte \wedge k < N$)



```

6       $C_{k+1} \leftarrow$  conjuntos de elementos candidatos generados a partir de  $L_k$ 
7      para cada transacción  $t$  en la base de datos  $re$  hacer
8          incrementar los recuentos de  $C_{k+1}$  contenida en  $t$ 
 $L_{k+1} \leftarrow$  candidatos en  $C_{k+1}$  que
9          satisfagan un apoyo mínimo  $\delta$ 
10          $k \leftarrow k + 1$ 
11     regreso  $\bigcup_k L_k$ 

```

El primer paso del algoritmo Apriori es identificar los conjuntos de elementos frecuentes comenzando con cada elemento de las transacciones que cumpla con el umbral de soporte mínimo predefinido. δ . Estos conjuntos de elementos son conjuntos de 1 elemento denotado como L_1 , ya que cada conjunto de 1 elemento contiene solo un elemento. A continuación, el algoritmo hace que los conjuntos de elementos se unan L_1 sobre sí mismo para formar nuevos conjuntos de 2 elementos, denotados como L_2 y determina el soporte de cada conjunto de 2 elementos en L_2 . Los conjuntos de elementos que no cumplen con el umbral mínimo de soporte δ son podados. El proceso de cultivo y poda se repite hasta que ningún conjunto de elementos alcance el umbral mínimo de apoyo. Opcionalmente, un límite *norte* se puede configurar para especificar el número máximo de elementos que puede alcanzar el conjunto de elementos o el número máximo de iteraciones del algoritmo. Una vez completado, la salida del algoritmo Apriori es la colección de todos los frecuentes k -conjuntos de elementos.

A continuación, se forma una colección de reglas candidatas en función de los conjuntos de elementos frecuentes que se descubren en el proceso iterativo descrito anteriormente. Por ejemplo, un conjunto de elementos frecuente { leche, huevos} puede sugerir reglas candidatas { leche} \rightarrow {huevos} y { huevos} \rightarrow {leche}.

5.3 Evaluación de las reglas candidatas

Los conjuntos de elementos frecuentes de la sección anterior pueden formar reglas candidatas como X implica Y ($X \rightarrow Y$). Esta sección analiza cómo medidas como la confianza, la elevación y el apalancamiento pueden ayudar a evaluar la idoneidad de estas reglas candidatas.

Confianza [2] se define como la medida de certeza o confiabilidad asociada con cada regla descubierta. Matemáticamente, la confianza es el porcentaje de transacciones que contienen tanto X como Y de todas las transacciones que contienen X (consulte la Ecuación 5-1).

$$\text{Confianza } (X \rightarrow Y) = \frac{\text{Soporte } (X \wedge Y)}{\text{Soporte } (X)} \quad (5-1)$$

Por ejemplo, si { pan, huevos, leche} tiene un soporte de 0.15 y { pan, huevos} también tiene un soporte de 0.15, la confianza de la regla { pan, huevos} \rightarrow {leche} es 1, lo que significa que el 100% del tiempo que un cliente compra pan y huevos, también compra leche. Por tanto, la regla es correcta para el 100% de las transacciones que contienen pan y huevos.

Una relación puede considerarse interesante cuando el algoritmo identifica la relación con una medida de confianza mayor o igual que un umbral predefinido. Este umbral predefinido se llama **confianza mínima**. Una mayor confianza indica que la regla ($X \rightarrow Y$) es más interesante o más confiable, según el conjunto de datos de muestra.

Hasta ahora, este capítulo ha hablado de dos medidas comunes que usa el algoritmo Apriori: soporte y confianza. Todas las reglas pueden clasificarse en función de estas dos medidas para filtrar las reglas poco interesantes y retener las interesantes.

Aunque la confianza puede identificar las reglas interesantes de todas las reglas candidatas, tiene un problema. Reglas dadas en forma de $X \rightarrow Y$, la confianza considera sólo el antecedente (X) y la concurrencia de X e Y ; no tiene en cuenta el consecuente de la regla (Y). Por lo tanto, la confianza no puede decir si una regla contiene una implicación verdadera de la relación o si la regla es pura coincidencia. X e Y pueden ser estadísticamente independientes y aun así recibir una puntuación de confianza alta. Otras medidas, como la elevación [6] y el apalancamiento [7], están diseñadas para abordar este problema.

Ascensor mide cuántas veces con más frecuencia X e Y ocurren juntas de lo esperado si son estadísticamente independientes entre sí. La elevación es una medida [6] de cómo X e Y están realmente relacionados en lugar de suceder casualmente juntos (ver Ecuación 5-2).

$$\text{Levantar } (X \rightarrow Y) = \frac{\text{Soporte } (XY)}{\text{Soporte } (X) * \text{Soporte } (Y)} \quad (5-2)$$

La elevación es 1 si X e Y son estadísticamente independientes entre sí. Por el contrario, una elevación de $X \rightarrow Y$ mayor que 1 indica que la regla tiene alguna utilidad. Un valor mayor de elevación sugiere una mayor fuerza de la asociación entre X e Y .

Suponiendo 1000 transacciones, con $\{\text{leche, huevos}\}$ apareciendo en 300 de ellos, $\{\text{Leche}\}$ apareciendo en 500 y $\{\text{huevos}\}$ apareciendo en 400, luego $\text{Levante } (\text{leche} \rightarrow \text{huevos}) = 0,3 / (0,5 * 0,4) = 1,5$. Si $\{\text{pan de molde}\}$ aparece en 400 transacciones y $\{\text{pan de leche}\}$ aparece en 400, luego $\text{Levante } (\text{leche} \rightarrow \text{pan}) = 0,4 / (0,5 * 0,4) = 2$. Por tanto, se puede concluir que la leche y el pan tienen una asociación más fuerte que la leche y los huevos.

Influencia [7] es una noción similar, pero en lugar de usar una razón, el apalancamiento usa la diferencia (ver Ecuación 5-3). El apalancamiento mide la diferencia en la probabilidad de que X e Y aparezcan juntos en el conjunto de datos en comparación con lo que se esperaría si X e Y fueran estadísticamente independientes entre sí.

$$\text{Apalancamiento } (X \rightarrow Y) = \text{Soporte } (X \wedge Y) - \text{Soporte } (X) * \text{Soporte } (Y) \quad (5-3)$$

En teoría, el apalancamiento es 0 cuando X e Y son estadísticamente independientes entre sí. Si X e Y tienen algún tipo de relación, el apalancamiento sería mayor que cero. Un valor de apalancamiento mayor indica una relación más fuerte entre X e Y . Para el ejemplo anterior, $\text{Apalancamiento } (\text{leche} \rightarrow \text{huevos}) = 0,3 - (0,5 * 0,4) = 0,1$ y $\text{Apalancamiento } (\text{leche} \rightarrow \text{pan}) = 0,4 - (0,5 * 0,4) = 0,2$. Nuevamente confirma que la leche y el pan tienen una asociación más fuerte que la leche y los huevos.

La confianza puede identificar reglas confiables, pero no puede decir si una regla es una coincidencia. Una regla de alta confianza a veces puede ser engañosa porque la confianza no considera consecuente el apoyo del conjunto de elementos en la regla. Medidas como la elevación y el apalancamiento no solo garantizan que se identifiquen reglas interesantes, sino que también filtran las reglas coincidentes.

Este capítulo ha analizado cuatro medidas de importancia e interés para las reglas de asociación: apoyo, confianza, impulso y apalancamiento. Estas medidas garantizan el descubrimiento de reglas interesantes y sólidas a partir de conjuntos de datos de muestra. Además de estas cuatro reglas, existen otras medidas alternativas, como la correlación [8], la fuerza colectiva [9], la convicción [6] y la cobertura [10]. Consulte la bibliografía para conocer cómo funcionan estas medidas.

5.4 Aplicaciones de las reglas de asociación

El término **análisis de la canasta de mercado** se refiere a una implementación específica de la minería de reglas de asociación que muchas empresas utilizan para una variedad de propósitos, incluidos estos:

- Enfoques a gran escala para mejorar la comercialización: qué productos deben incluirse o excluirse del inventario cada mes.
- Comercialización cruzada entre productos y artículos de alto margen o alto precio
- Colocación física o lógica del producto dentro de categorías relacionadas de productos.
- Programas promocionales: múltiples incentivos de compra de productos gestionados a través de un programa de tarjetas de fidelización.

Además del análisis de la cesta de la compra, las reglas de asociación se utilizan comúnmente para los sistemas de recomendación [11] y el análisis de flujo de clics [12].

Muchos proveedores de servicios en línea como Amazon y Netlix utilizan sistemas de recomendación. Los sistemas de recomendación pueden usar reglas de asociación para descubrir productos relacionados o identificar clientes que tengan intereses similares. Por ejemplo, las reglas de asociación pueden sugerir que los clientes que han comprado el producto A también han comprado el producto B, o que los clientes que han comprado los productos A, B y C son más similares a este cliente. Estos hallazgos brindan oportunidades para que los minoristas realicen ventas cruzadas de sus productos.

El análisis del flujo de clics se refiere al análisis de los datos relacionados con la navegación web y los clics de los usuarios, que se almacenan en el lado del cliente o del servidor. Los archivos de registro de uso de la web generados en servidores web contienen una gran cantidad de información, y las reglas de asociación pueden proporcionar conocimientos útiles a los analistas de datos de uso de la web. Por ejemplo, las reglas de asociación pueden sugerir que los visitantes del sitio web que llegan a la página X hacen clic en los enlaces A, B y C con mucha más frecuencia que en los enlaces D, E y F. Esta observación proporciona información valiosa sobre cómo personalizar y recomendar mejor el contenido a los visitantes del sitio.

La siguiente sección muestra un ejemplo de transacciones de una tienda de comestibles y demuestra cómo usar R para realizar la extracción de reglas de asociación.

5.5 Un ejemplo: transacciones en una tienda de comestibles

Un ejemplo ilustra la aplicación del algoritmo Apriori a un caso relativamente simple que se generaliza a los utilizados en la práctica. Usando R y el arules y arulesViz paquetes, este ejemplo muestra cómo utilizar el algoritmo Apriori para generar conjuntos de elementos y reglas frecuentes y para evaluar y visualizar las reglas.

Los siguientes comandos instalan estos dos paquetes y los importan al espacio de trabajo actual de R:

```
install.packages ('arules')
install.packages ('arulesViz')

biblioteca ('arules')
biblioteca ('arulesViz')
```

5.5.1 El conjunto de datos de comestibles

El ejemplo usa el **Comestibles** conjunto de datos de la R arules paquete. los **Comestibles** El conjunto de datos se recopila a partir de 30 días de transacciones en el punto de venta del mundo real de una tienda de comestibles. El conjunto de datos contiene 9,835 transacciones, y los elementos se agregan en 169 categorías.

datos (comestibles)

Comestibles

transacciones en formato disperso con 9835

transacciones (filas) y 169 elementos (columnas)

El resumen muestra que los elementos más frecuentes en el conjunto de datos incluyen elementos como leche entera, otras verduras, panecillos / bollos, refrescos y yogur. Estos artículos se compran con más frecuencia que los demás.

resumen (Comestibles)

transacciones como itemMatrix en formato disperso con 9835 filas

(elementos / conjuntos de elementos / transacciones) y 169 columnas

(elementos) y una densidad de 0.02609146

artículos más frecuentes:

leche entera otras verduras		rollos / bollos	soda
2513	1903	1809	1715
yogur	(Otro)		
1372	34055		

distribución de la longitud del elemento (conjunto de elementos / transacción): tamaños

1	2	3	4	5	6	7	8	9	10	11	12	13	14	
2159	1643	1299	1005		855	645	545	438	350	246	182	117	78	77
15	diecisés	17	18	19	20	21	22	23	24	26	27	28	29	
55	46	29	14	14	9	11	4	6	1	1	1	1	3	
32														
1														
Min. 1er Qu. Mediana				Media 3 ^a Qu.				Max.						
1.000	2.000	3.000		4.409	6.000	32.000								

incluye información ampliada del artículo; ejemplos:

etiquetas level2 nivel 1

1 salchicha de Frankfurt y salchicha 2

salchicha salchicha y salchicha 3 pan de hígado

salchicha y salchicha

La clase del conjunto de datos es **actas**, como lo define el arules paquete. los actas La clase contiene tres espacios:

- **transactionInfo**: Marco Adata con vectores de la misma longitud que el número de transacciones
- **itemInfo**: Marco Adata para almacenar etiquetas de artículos
- **datos**: Matriz de incidencia binaria que indica qué etiquetas de artículo aparecen en cada transacción

```
class (Comestibles)
[1] "transacciones"
attr(, "paquete")
[1] "arules"
```

Para el **Comestibles** conjunto de datos, el transactionInfo no se está utilizando. Entrar

Comestibles @ itemInfo para mostrar las 169 etiquetas de comestibles, así como sus categorías. El siguiente comando muestra solo las primeras 20 etiquetas de comestibles. Cada etiqueta de comestibles se asigna a dos niveles de categorías: nivel 2 y nivel 1 -dónde nivel 1 es un superconjunto de nivel 2. Por ejemplo, etiqueta de comestibles salchicha pertenece a la salchicha categoría en nivel 2, y es parte del carne y embutidos categoría en nivel 1. (Tenga en cuenta que " reunirse "En nivel 1 es un error tipográfico en el conjunto de datos).

Comestibles @ itemInfo [1:20,]

	etiquetas	nivel 2	nivel 1
1	salchicha	salchicha	conocer y salchicha
2	salchicha	salchicha	conocer y salchicha
3	pan de hígado	salchicha	conocer y salchicha
4	jamón	salchicha	conocer y salchicha
5	carne	salchicha	conocer y salchicha
6	productos terminados	salchicha	conocer y salchicha
7	salchicha orgánica	salchicha	conocer y salchicha
8	pollo	aves de corral	conocer y salchicha
9	pavo	aves de corral	conocer y salchicha
10	Cerdo	Cerdo	conocer y salchicha
11	carne de vaca	carne de vaca	conocer y salchicha
12	carne de hamburguesa	carne de vaca	conocer y salchicha
13	pescado	pescado	conocer y salchicha
14	Fruta cítrica	frutas frutas y verduras frutas frutas y	
15	fruta tropical	verduras frutas frutas y verduras frutas	
dieciséis	pepita de fruta	frutas y verduras frutas frutas y verduras	
17	uvas	frutas frutas y verduras	
18	bayas		
19	nueces / ciruelas pasas		
20	hortalizas de raíz hortalizas frutas y hortalizas		

El siguiente código muestra las transacciones 10 a 20 del Comestibles conjunto de datos. los [10:20] se puede cambiar a [1: 9835] para mostrar todas las transacciones.

```
aplicar ( Groceries @ data [, 10:20], 2,
  función (r) pegar ( Comestibles @ itemInfo [r, "etiquetas"], contraer = ",")
  )
```

Cada fila de la salida muestra una transacción que incluye uno o más productos, y cada transacción corresponde a todo en el carrito de compras de un cliente. Por ejemplo, en la primera transacción, un cliente compró leche entera y cereales.

- [1] "leche entera, cereales"
- [2] "frutas tropicales, otras verduras, pan blanco, agua embotellada, chocolate"
- [3] "cítricos, frutas tropicales, leche entera, mantequilla, cuajada, yogur, harina, agua embotellada, platos"
- [4] "carne de res"

- [5] "salchicha, panecillos, refrescos" [6] "pollo, frutas tropicales"
- [7] "mantequilla, azúcar, jugo de frutas / verduras, periódicos" [8] "jugo de frutas / verduras"
- [9] "frutas / hortalizas envasadas"
- [10] "chocolate"
- [11] "barra de especialidades"

La siguiente sección muestra cómo generar conjuntos de elementos frecuentes a partir de **Comestibles** conjunto de datos.

5.5.2 Generación frecuente de conjuntos de elementos

los a priori() función de la una regla El paquete implementa el algoritmo Apriori para crear conjuntos de elementos frecuentes. Tenga en cuenta que, de forma predeterminada, a priori() La función ejecuta todas las iteraciones a la vez. Sin embargo, para ilustrar cómo funciona el algoritmo Apriori, los ejemplos de código de esta sección establecen manualmente los parámetros del a priori() función para simular cada iteración del algoritmo.

Suponga que el umbral de soporte mínimo se establece en 0.02 según la discreción de la administración. Debido a que el conjunto de datos contiene 9 853 transacciones, un conjunto de elementos debe aparecer al menos 198 veces para que se considere un conjunto de elementos frecuente. La primera iteración del algoritmo Apriori calcula el soporte de cada producto en el conjunto de datos y retiene aquellos productos que satisfacen el soporte mínimo. El siguiente código identifica 59 conjuntos frecuentes de 1 elemento que satisfacen el soporte mínimo. Los parámetros de a priori()

especifique las longitudes mínima y máxima de los conjuntos de elementos, el umbral de soporte mínimo y el objetivo que indica el tipo de asociación extraída.

```
itemsets <- apriori (Comestibles, parámetro = lista (minlen = 1, maxlen = 1,
support = 0.02, target = "conjuntos de elementos frecuentes"))
```

especificación de parámetros:

confianza	minval	smax	arem	aval	original	Soporte	soporte	minlen					
0,8	0,1		1	ninguno	FALSO				CIERTO	0,02			1
Maxlen			objetivo		ext								
1 conjuntos de elementos frecuentes FALSO													

control algorítmico:

filtrar	árbol	montón	memopt	cargar	ordenar	detallado
0,1	VERDADERO	VERDADERO	FALSO	VERDADERO	2	CIERTO

apriori - encuentra reglas de asociación con el algoritmo apriori versión 4.21 (2004.05.09)

(c) 1996-2004 Christian Borgelt

establecer apariciones de elementos ... [0 elemento (s)] hecho [0.00s].

establecer transacciones ... [169 artículo (s), 9835 transacción (s)] realizada [0.00s]. clasificación y recodificación de elementos ... [59 elemento (s)] realizado [0.00s]. creando árbol de transacciones ... hecho [0.00s].

comprobando subconjuntos de tamaño 1 hecho [0.00s].

escribiendo ... [59 conjunto (s)] hecho [0.00s]. creando objeto

S4 ... hecho [0.00s].

El resumen de los conjuntos de elementos muestra que el soporte de conjuntos de 1 elemento varía entre 0,02105 y 0,25552. Debido a que el soporte máximo de los conjuntos de 1 elemento en el conjunto de datos es solo 0,25552, para permitir el descubrimiento de reglas interesantes, el umbral de soporte mínimo no debe establecerse demasiado cerca de ese número.

resumen (conjuntos de elementos)
conjunto de 59 elementos

artículos más frecuentes:

salchicha	salchicha	jamón	carme	pollo
1	1	1	1	1
(Otro)				
54				

distribución de la longitud del elemento (conjunto de elementos / transacción): tamaños 1

59

Min. 1er Qu. Mediana	Media 3 ^a Qu.	Max.
1	1	1

resumen de medidas de calidad:

apoyo		
Min.	: 0.02105	
1er Qu.:	0.03015	
Mediana:	0.04809	
Media	: 0.06200	
3er Qu.:	0.07666	
Max.	: 0.25552	

incluye listas de ID de transacciones: FALSE

información minera:

ntransacciones de datos respaldan la confianza Comestibles		
9835	0,02	1

El siguiente código usa el inspeccionar() función para mostrar los 10 principales conjuntos de elementos 1 frecuentes ordenados por su soporte. De todos los registros de transacciones, los 59 conjuntos de 1 elemento como {leche entera}, {otras verduras}, {panecillos / bollos}, {refresco}, y {yogur} todos satisfacen el mínimo apoyo. Por lo tanto, se denominan conjuntos frecuentes de 1 elemento.

inspeccionar (head (sort (itemsets, by = "support"), 10))

	articulos	apoyo
1	{leche entera}	0,25551601
2	{otras verduras}	0,19349263
3	{panecillos / bollos}	0,18393493
4	{soda}	0,17437722
5	{yogur}	0,13950178
6	{agua embotellada}	0,11052364

7 {hortalizas de raíz}	0.10899847
8 {fruta tropical}	0.10493137
9 {bolsas de compras}	0.09852567
10 {salchicha}	0.09395018

En la siguiente iteración, la lista de conjuntos de 1 elementos frecuentes se une a sí misma para formar todos los conjuntos de 2 elementos candidatos posibles. Por ejemplo, 1-itemsets { leche entera} y { soda} se uniría para convertirse en un conjunto de 2 elementos { leche entera, refresco}. El algoritmo calcula el soporte de cada conjunto candidato de 2 elementos y retiene aquellos que satisfacen el soporte mínimo. El resultado que sigue muestra que se han identificado 61 conjuntos frecuentes de 2 elementos.

```
itemsets <- apriori (Comestibles, parámetro = lista (minlen = 2, maxlen = 2,
support = 0.02, target = "conjuntos de elementos frecuentes"))
```

especificación de parámetros:

confianza	minval	smax	arem	aval	original	Soporte	soporte	minlen
0,8	0,1		1	ninguno	FALSO			
Maxlen			objetivo		ext			
						CIERTO	0,02	2
							2 conjuntos de elementos frecuentes	FALSO

control algorítmico:

filtrar	árbol	montón	memopt	cargar	ordenar	detallado
0,1	VERDADERO	VERDADERO	FALSO	VERDADERO	2	CIERTO

apriori - encuentra reglas de asociación con el algoritmo apriori versión 4.21 (2004.05.09)

(c) 1996-2004 Christian Borgelt

establecer apariciones de elementos ... [0 elemento (s)] hecho [0.00s].

establecer transacciones ... [169 artículo (s), 9835 transacción (s)] realizada [0.00s]. clasificación y recodificación de elementos ... [59 elemento (s)] realizado [0.00s]. creando árbol de transacciones ... hecho [0.00s].

comprobando subconjuntos de tamaño 1 2 hecho [0.00s].

escribiendo ... [61 conjunto (s)] hecho [0.00s]. creando objeto S4 ... hecho [0.00s].

El resumen de los conjuntos de elementos muestra que el soporte de conjuntos de 2 elementos varía de 0.02003 a 0.07483.

resumen (conjuntos de elementos)

conjunto de 61 elementos

artículos más frecuentes:

leche entera	otras verduras	yogur	rollos / bollos
25	17	9	9
soda	(Otro)		
9	53		

distribución de la longitud del elemento (conjunto de elementos / transacción): tamaños 2

Min.	1er Qu.	Mediana		Media 3 ^a Qu.		Max.
2	2	2		2		2

resumen de medidas de calidad:

apoyo

Min. : 0.02003

1er Qu.:0.02227

Mediana: 0.02613

Media : 0.02951

3er Qu.:0.03223

Max. : 0.07483

incluye listas de ID de transacciones: FALSE

información minera:

ntransacciones de datos respaldan la confianza Comestibles

9835	0,02	1
------	------	---

A continuación, se muestran los 10 conjuntos de 2 elementos más frecuentes, ordenados por su soporte. Observe que la leche entera aparece seis veces en los 10 principales conjuntos de 2 elementos clasificados por apoyo. Como se vio anteriormente, { leche entera} tiene el mayor soporte entre todos los conjuntos de 1 elemento. Estos 10 conjuntos de 2 elementos principales con el mayor soporte pueden no ser interesantes; esto resalta las limitaciones de usar el soporte solo.

inspeccionar (head (sort (itemsets, by = "support"), 10))

	artículos	apoyo
1	{otras verduras, leche entera}	0.07483477
2	{leche entera, rollos / bollos}	0.05663447
3	{leche entera, yogur}	0.05602440
4	{tubérculos, leche entera}	0.04890696
5	{tubérculos, otras verduras}	0.04738180
6	{otras verduras, yogur}	0.04341637
7	{otras verduras, rollos / bollos}	0.04260295
8	{fruta tropical, leche entera}	0.04229792
9	{leche entera, soda}	0.04006101
10	{panecillos / bollos, soda}	0.03833249

A continuación, la lista de conjuntos de 2 elementos frecuentes se une a sí misma para formar conjuntos de 3 elementos candidatos. Por ejemplo {otras verduras, leche entera} y { leche entera, panecillos / bollos} se uniría como { otras verduras, leche entera, panecillos / bollos}. El algoritmo conserva esos conjuntos de elementos

que satisfagan el mínimo apoyo. El siguiente resultado muestra que solo se han identificado dos conjuntos frecuentes de 3 elementos.

```
itemsets <- apriori (Comestibles, parámetro = lista (minlen = 3, maxlen = 3,
support = 0.02, target = "conjuntos de elementos frecuentes"))
```

especificación de parámetros:

confianza minval smax arem aval original Soporte soporte minlen	0,8	0,1	1 ninguno FALSO	CIERTO	0,02	3
Maxlen		objetivo	ext			
3 conjuntos de elementos frecuentes FALSO						

control algorítmico:

filtrar árbol montón memopt cargar ordenar detallado	0,1 VERDADERO VERDADERO FALSO VERDADERO	2	CIERTO
--	---	---	--------

apriori - encuentra reglas de asociación con el algoritmo apriori versión 4.21 (2004.05.09)
(c) 1996-2004 Christian Borgelt

establecer apariciones de elementos ... [0 elemento (s)] hecho [0.00s].
establecer transacciones ... [169 artículo (s), 9835 transacción (s)] realizada [0.00s]. clasificación y
recodificación de elementos ... [59 elemento (s)] realizado [0.00s]. creando árbol de transacciones ... hecho
[0.00s].

comprobando subconjuntos de tamaño 1 2 3 hecho [0.00s].
escribiendo ... [2 conjunto (s)] hecho [0.00s]. creando objeto S4 ...
hecho [0.00s].

Los conjuntos de 3 elementos se muestran a continuación:

inspeccionar (ordenar (conjuntos de elementos, por = "apoyo"))	
artículos	apoyo
1 {tubérculos,	
otras verduras,	
leche entera}	0.02318251
2 {otras verduras,	
leche entera,	
yogur}	0.02226741

En la siguiente iteración, solo hay un candidato de 4 elementos
{tubérculos, otras verduras, leche entera, yogur}, y su soporte está por debajo de 0,02. No se han encontrado
conjuntos de 4 elementos frecuentes y el algoritmo converge.

```
itemsets <- apriori (Comestibles, parámetro = lista (minlen = 4, maxlen = 4,
support = 0.02, target = "conjuntos de elementos frecuentes"))
```

especificación de parámetros:

confianza minval smax arem aval original Soporte soporte minlen	0,8	0,1	1 ninguno FALSO	CIERTO	0,02	4
Maxlen		objetivo	ext			
4 conjuntos de elementos frecuentes FALSO						

control algorítmico:

filtrar árbol montón memopt cargar ordenar detallado
 0.1 VERDADERO VERDADERO FALSO VERDADERO 2 CIERTO

apriori - encuentra reglas de asociación con el algoritmo apriori versión 4.21 (2004.05.09)

(c) 1996-2004 Christian Borgelt

establecer apariciones de elementos ... [0 elemento (s)] hecho [0.00s].

establecer transacciones ... [169 artículo (s), 9835 transacción (s)] realizada [0.00s]. clasificación y recodificación de elementos ... [59 elemento (s)] realizado [0.00s]. creando árbol de transacciones ... hecho [0.00s].

comprobando subconjuntos de tamaño 1 2 3 hecho [0.00s].

escribiendo ... [0 conjunto (s)] hecho [0.00s]. creando objeto S4 ... hecho [0.00s].

Los pasos anteriores simulan el algoritmo Apriori en cada iteración. Para el **Comestibles** conjunto de datos, las iteraciones se quedan sin soporte cuando $k = 4$. Por lo tanto, los conjuntos de elementos frecuentes contienen 59 conjuntos frecuentes de 1 elemento, 61 conjuntos frecuentes de 2 elementos y 2 conjuntos frecuentes de 3 elementos.

Cuando el Maxlen El parámetro no está configurado, el algoritmo continúa cada iteración hasta que se agota el soporte o hasta que k alcanza el valor predeterminado maxlen = 10. Como se muestra en el resultado del código que sigue, se han identificado 122 conjuntos de elementos frecuentes. Esto coincide con el número total de 59 conjuntos frecuentes de 1 elemento, 61 conjuntos frecuentes de 2 elementos y 2 conjuntos frecuentes de 3 elementos.

itemsets <- apriori (Comestibles, parámetro = lista (minlen = 1, soporte = 0.02,

target = "conjuntos de elementos frecuentes")

especificación de parámetros:

confianza minval smax arem aval original Soporte soporte minlen	0,8	0,1	1 ninguno FALSO		CIERTO	0,02	1
Maxlen		objetivo	ext				
10 conjuntos de elementos frecuentes FALSO							

control algorítmico:

filtrar árbol montón memopt cargar ordenar detallado
 0.1 VERDADERO VERDADERO FALSO VERDADERO 2 CIERTO

apriori - encuentra reglas de asociación con el algoritmo apriori versión 4.21 (2004.05.09)

(c) 1996-2004 Christian Borgelt

establecer apariciones de elementos ... [0 elemento (s)] hecho [0.00s].

establecer transacciones ... [169 artículo (s), 9835 transacción (s)] realizada [0.00s]. clasificación y recodificación de elementos ... [59 elemento (s)] realizado [0.00s]. creando árbol de transacciones ... hecho [0.00s].

comprobando subconjuntos de tamaño 1 2 3 hecho [0.00s].

escribiendo ... [122 conjunto (s)] hecho [0.00s]. creando objeto S4 ... hecho [0.00s].

Tenga en cuenta que los resultados se evalúan en función del contexto empresarial específico del ejercicio utilizando el conjunto de datos específico. Si el conjunto de datos cambia o se elige un umbral de soporte mínimo diferente, el algoritmo Apriori debe ejecutar cada iteración nuevamente para recuperar los conjuntos de elementos frecuentes actualizados.

5.5.3 Generación y visualización de reglas

los a priori() La función también se puede utilizar para generar reglas. Suponga que el umbral mínimo de apoyo ahora está establecido en un valor más bajo de 0.001 y el umbral mínimo de confianza está establecido en 0.6. Un umbral de soporte mínimo más bajo permite que aparezcan más reglas. El siguiente código crea 2918 reglas a partir de todas las transacciones en el *Comestibles* conjunto de datos que satisfaga tanto el soporte mínimo como la confianza mínima.

```
reglas <- apriori (Comestibles, parámetro = lista (soporte = 0.001,
confianza = 0,6, objetivo = "reglas"))
```

especificación de parámetros:

confianza minval smax arem aval original Soporte soporte minlen	0,6	0,1	1 ninguno FALSO	CIERTO	0,001	1
objetivo maxlen ext	10 reglas	FALSO				

control algorítmico:

filtrar árbol montón memopt cargar ordenar detallado	1 VERDADERO VERDADERO FALSO VERDADERO	2	CIERTO
--	---------------------------------------	---	--------

apriori - encuentra reglas de asociación con el algoritmo apriori versión 4.21 (2004.05.09)

(c) 1996-2004 Christian Borgelt

establecer apariciones de elementos ... [0 elemento (s)] hecho [0.00s].

establecer transacciones ... [169 artículo (s), 9835 transacción (s)] realizada [0.00s]. clasificación y recodificación de elementos ... [157 elemento (s)] realizado [0.00s]. creando árbol de transacciones ... hecho [0.00s].

comprobando subconjuntos de tamaño 1 2 3 4 5 6 hecho [0.01s]. escribiendo ...

[2918 regla (s)] hecho [0.00s]. creando objeto S4 ... hecho [0.01s].

El resumen de las reglas muestra el número de reglas y rangos de soporte, confianza y elevación.

resumen (reglas)

conjunto de 2918 reglas

distribución de la longitud de la regla (lhs + rhs): tamaños

2	3	4	5	6
3 490	1765	626		34

Min. 1er Qu. Mediana	Media 3 ^a Qu.	Max.
2.000	4.000	4.000
	4.068	4.000
		6.000

resumen de medidas de calidad:

apoyo	confianza	ascensor
Min. : 0,001017	Min. : 0,6000	Min. : 2.348
1er Qu.:0,001118	1er Qu.:0,6316	1er Qu.: 2.668
Mediana: 0,001220	Mediana: 0,6818	Mediana: 3,168
Media : 0,001480	Media : 0,7028	Media : 3,450
3er Qu.:0,001525	3er Qu.:0,7500	3er Qu.: 3,692
Max. : 0,009354	Max. : 1.0000	Max. : 18.996

información minera:

ntransacciones de datos respaldan la confianza Comestibles
9835 0,001 0,6

Entrar trama (reglas) para mostrar el diagrama de dispersión de las 2918 reglas (Figura 5-3), donde el eje horizontal es el soporte, el eje vertical es la confianza y el sombreado es el ascensor. El diagrama de dispersión muestra que, de los 2.918 ru

y una baja confianza

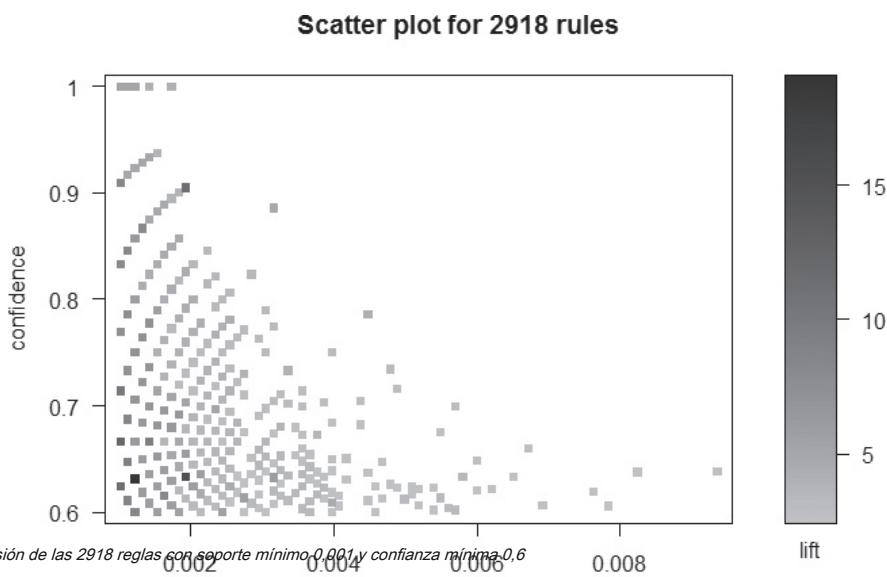


FIGURE 5-3 Diagrama de dispersión de las 2918 reglas con soporte mínimo 0,001 y confianza mínima 0,6

Entrando trama (reglas @ calidad) muestra una matriz de gráficos de dispersión (Figura 5-4) para comparar el apoyo, la confianza y la elevación de las 2918 reglas.

La figura 5-4 muestra que la sustentación es proporcional a la confianza e ilustra varios agrupamientos lineales. Como lo indican la Ecuación 5-2 y la Ecuación 5-3, $Elevación = Confianza / Soporte (Y)$. Por lo tanto, cuando el soporte de Y permanece igual, la elevación es proporcional a la confianza, y la pendiente de la tendencia lineal es la inversa.

cal de Soporte (Y). El siguiente código muestra que, de las 2918 reglas, solo hay 18 valores diferentes para

1 , y la mayoría ocurre en las pendientes 3.91, 5.17, 7.17, 9.17 y 9.53. Esto coincide con las pistas mostradas Soporte (Y) en la tercera columna y la segunda columna de la Figura 5-4, donde el eje x es la confianza y el eje y es la sustentación.

```
# calcula el 1 / Soporte (Y)
pendiente <- ordenar (ronda ( reglas @ calidad $ elevación / reglas @ calidad $ confianza, 2))
# Muestra el número de veces que aparece cada pendiente en el conjunto de datos
unlist (lapply (split (pendiente, f = pendiente), longitud))
```

3,91 5,17 5,44 5,73 7,17 9,05 9,17 9,53 10,64 12,08 1585
 940 12 7 188 1 102 55 1 4

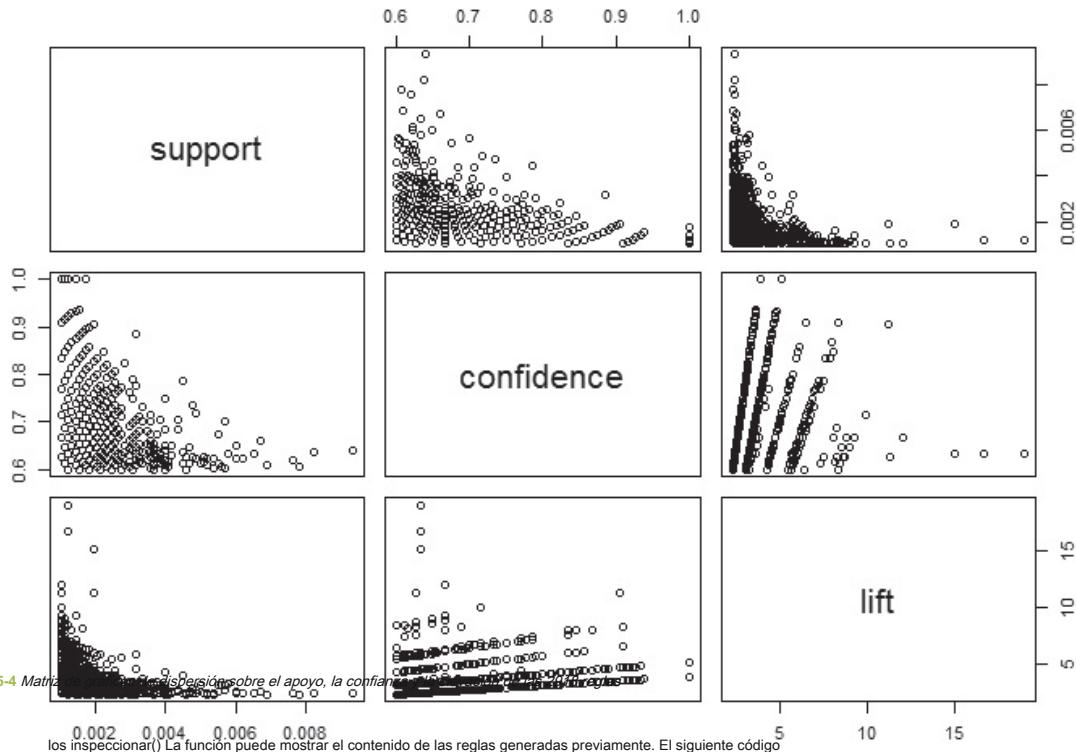


FIGURE 5-4 Matriz de dispersión de las reglas generadas. La figura muestra la dispersión sobre el apoyo, la confianza y el alza.

Los resultados se presentan en una matriz de dispersión. La función puede mostrar el contenido de las reglas generadas previamente. El siguiente código

muestra las diez reglas principales ordenadas por elevación. Regla { Productos alimenticios instantáneos, refrescos } → {carne de hamburguesa} tiene la elevación más alta de 18,995654.

```
inspeccionar (cabeza (ordenar (reglas, por = "levantar"), 10))
lhs          rhs
apoyo      confianza      ascensor
1 {Inst   productos alimenticios para hormigas,
    soda }           => {carne de hamburguesa}
0. 001220 132  0,6315789 18,995654
2 {soda ,
    popc orn}        => {bocadillo salado}
0. 001220 132  0,6315789 16,697793
3 {jamón,
    proc esse d queso}      => {pan blanco}
0. 001931 876  0,6333333 15,045491
```

```

4 {fruta tropical,
  otras verduras,
  yogur,
  pan blanco} => {mantequilla}
0. 001016777 0,6666667 12,030581
5 {carne de hamburguesa,
  yogur,
  azotado/ cCrea agria} => {mantequilla}
0. 001016777 0,6250000 11,278670
6 {tropical Fruta,
  otras ve getables,
  mi entera lk,
  yogur,
  Doméstico huevos} => {mantequilla}
0. 001016777 0,6250000 11,278670
7 {espíritu,
  rojo / rubor h vino} => {cerveza embotellada}
0. 001931876 0,9047619 11,235269
8 {otro ve getables,
  mantequilla,
  azúcar} => {crema batida / agria}
0. 001016777 0,7142857 9,964539
9 {mi entera lk,
  mantequilla,
  duro che ese} => {crema batida / agria}
0. 001423488 0,6666667 9,300236
10 {tropical Fruta,
  otras ve getables,
  mantequilla,
  jugo de frutas / verduras} => {crema batida / agria}
0. 001016777 0,6666667 9,300236

```

El siguiente código obtiene un total de 127 reglas cuya confianza está por encima de 0.9:

```

trustRules <- reglas [calidad (reglas) $ confianza> 0.9]
reglas de confianza
conjunto de 127 reglas

```

El siguiente comando produce una visualización basada en matrices (Figura 5-5) del LHS frente al RHS de las reglas. La leyenda de la derecha es una matriz de colores que indica la elevación y la confianza a la que corresponde cada cuadrado de la matriz principal.

```

trama (reglas confiables, método = "matriz", medida = c ("elevación", "confianza"),
control = lista (reordenar = VERDADERO))

```

Como el anterior trama() ejecute el comando, la consola R mostraría simultáneamente una lista distinta de LHS y RHS de las 127 reglas. Aquí se muestra un segmento de la salida:

```

Conjuntos de elementos en Antecedente (LHS)
[1] "[cítricos, otras verduras, refrescos, zumos de frutas / verduras]" [2] "[frutas tropicales, otras
verduras, leche entera, yogur, aceite]"

```

[3] "frutas tropicales, mantequilla, crema batida / agria, jugo de frutas / verduras"

[4] "fruta tropical, uvas, leche entera, yogur" [5] "jamón, fruta tropical, pepitas de fruta, leche entera"

...

[124] "licor, vino tinto / colorete"

Conjuntos de elementos en Consequent (RHS) [1]

"leche entera"

"yogur"

"tubérculos"

[4] "cerveza embotellada"

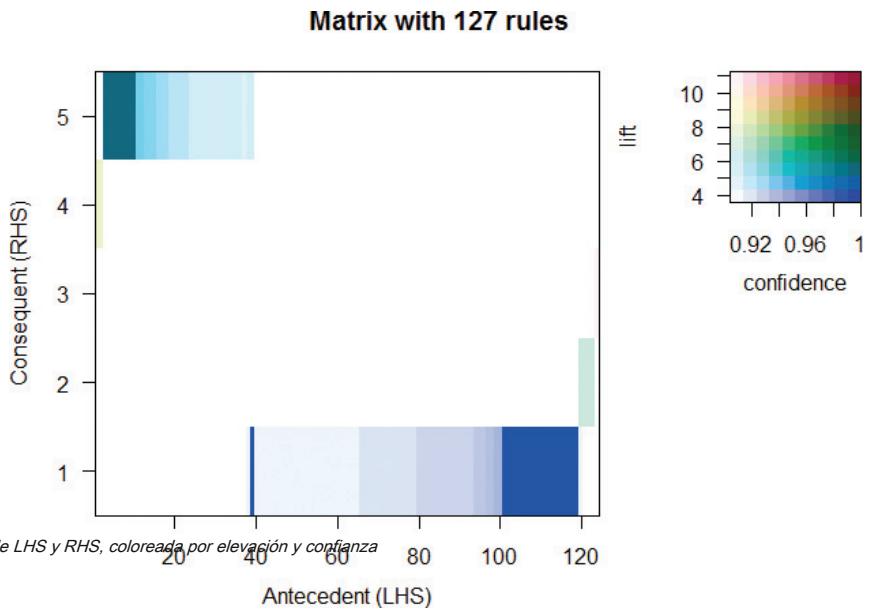


FIGURE 5-5 Visualización matricial de LHS y RHS, coloreada por elevación y confianza

El siguiente código proporciona una visualización de las cinco reglas principales con la mayor elevación. La gráfica se muestra en la Figura 5-6. En el gráfico, la flecha siempre apunta de un elemento en el lado izquierdo a un elemento en el lado derecho. Por ejemplo, las flechas que conectan jamón, queso procesado y pan blanco sugieren una regla

{jamón, queso fundido} → {pan blanco}. La leyenda en la parte superior derecha del gráfico muestra que el tamaño de un círculo indica el apoyo de las reglas que van de 0,001 a 0,002. El color (o sombra) representa la elevación, que varía de 11,279 a 18,996. La regla con mayor elevación es {Productos alimenticios instantáneos, refrescos} → {carne de hamburguesa}.

```
highLiftRules <- head (sort (rules, by = "lift"), 5)
plot (highLiftRules, method = "graph", control = list (type = "items"))
```

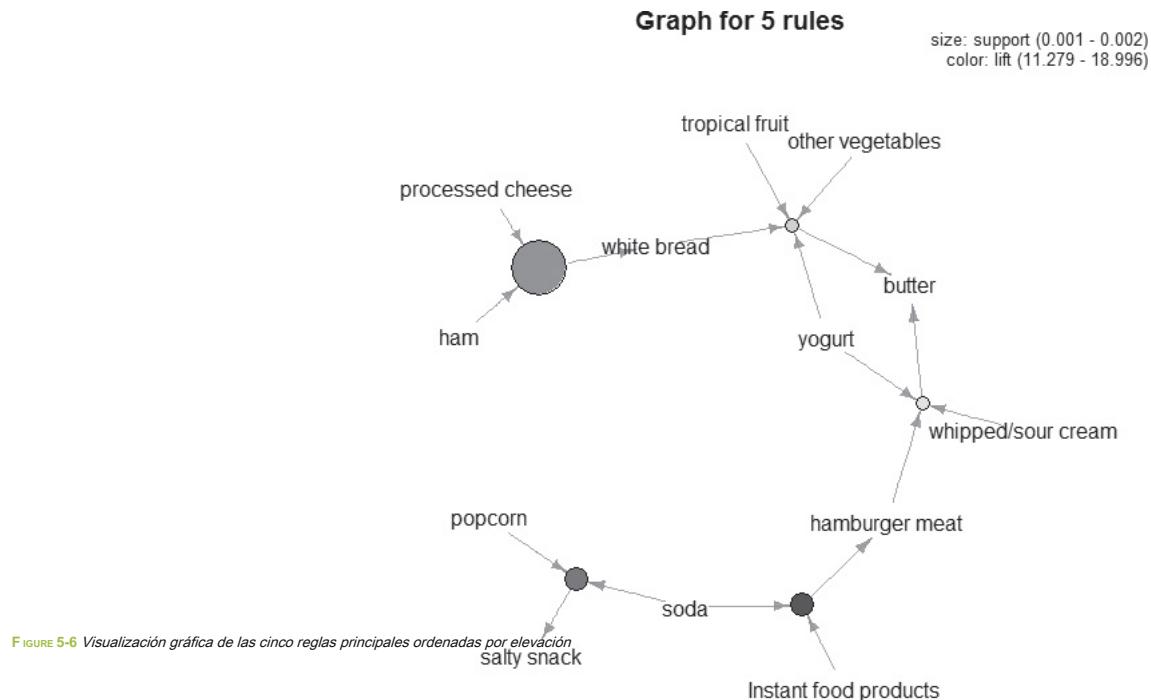


FIGURE 5-6 Visualización gráfica de las cinco reglas principales ordenadas por elevación

5.6 Validación y prueba

Después de recopilar las reglas de salida, puede que sea necesario utilizar uno o más métodos para validar los resultados en el contexto empresarial para el conjunto de datos de muestra. El primer enfoque se puede establecer mediante medidas estadísticas como la confianza, la elevación y el apalancamiento. Las reglas que involucran elementos mutuamente independientes o cubren pocas transacciones se consideran poco interesantes porque pueden capturar relaciones espirituales.

Como se mencionó en la Sección 5.3, la confianza mide la posibilidad de que X e Y aparezcan juntos en relación con la posibilidad de que X aparezca. La confianza se puede utilizar para identificar el interés de las reglas.

Tanto el levantamiento como el apalancamiento comparan el apoyo de X e Y con su apoyo individual. Mientras se extraen datos con reglas de asociación, algunas reglas generadas podrían ser pura coincidencia. Por ejemplo, si el 95% de los clientes compran X y el 90% de los clientes compran Y, entonces X e Y ocurrirían juntos al menos el 85% del tiempo, incluso si no hay relación entre los dos. Medidas como el impulso y el apalancamiento garantizan que se identifiquen reglas interesantes en lugar de coincidentes.

Se puede establecer otro conjunto de criterios mediante argumentos subjetivos. Incluso con una gran confianza, una regla puede considerarse subjetivamente poco interesante a menos que revele acciones rentables inesperadas. Por ejemplo, reglas como { papel } → { lápiz } puede no ser subjetivamente interesante o significativo a pesar de los altos valores de apoyo y confianza. Por el contrario, una regla como { pañal } → { cerveza } que satisfaga tanto el apoyo mínimo como la confianza mínima puede considerarse subjetivamente interesante porque esta regla es

inesperado y puede sugerir una oportunidad de venta cruzada para el minorista. Esta incorporación de conocimiento subjetivo en la evaluación de reglas puede ser una tarea difícil y requiere la colaboración de expertos en el dominio. Como se vio en el Capítulo 2, "Ciclo de vida del análisis de datos", los expertos del dominio pueden actuar como usuarios comerciales o como analistas de inteligencia comercial como parte del equipo de Ciencia de datos. En la Fase 5, el equipo puede comunicar los resultados y decidir si es apropiado ponerlos en funcionamiento.

5.7 Diagnóstico

Aunque el algoritmo Apriori es fácil de entender e implementar, algunas de las reglas generadas son poco interesantes o prácticamente inútiles. Además, algunas de las reglas pueden generarse debido a relaciones coincidentes entre las variables. Medidas como la confianza, la sustentación y el apalancamiento deben usarse junto con las percepciones humanas para abordar este problema.

Otro problema con las reglas de asociación es que, en la Fase 3 y 4 del Ciclo de vida de análisis de datos (Capítulo 2), el equipo debe especificar el soporte mínimo antes de la ejecución del modelo, lo que puede llevar a demasiadas o muy pocas reglas. En investigaciones relacionadas, una variante del algoritmo [13] puede usar un rango objetivo predefinido para el número de reglas, de modo que el algoritmo pueda ajustar el soporte mínimo en consecuencia.

La Sección 5.2 presentó el algoritmo Apriori, que es uno de los primeros y más fundamentales algoritmos para generar reglas de asociación. El algoritmo Apriori reduce la carga de trabajo computacional al examinar solo los conjuntos de elementos que cumplen con el umbral mínimo especificado. Sin embargo, dependiendo del tamaño del conjunto de datos, el algoritmo Apriori puede resultar computacionalmente costoso. Para cada nivel de soporte, el algoritmo requiere un escaneo de toda la base de datos para obtener el resultado. En consecuencia, a medida que la base de datos crece, se necesita más tiempo para calcular en cada ejecución. A continuación, se muestran algunos enfoques para mejorar la eficiencia de Apriori:

- **Fraccionamiento:** Cualquier conjunto de elementos que sea potencialmente frecuente en una base de datos de transacciones debe ser frecuente en al menos una de las particiones de la base de datos de transacciones.
- **Muestreo:** Esto extrae un subconjunto de los datos con un umbral de soporte más bajo y usa el subconjunto para realizar la minería de reglas de asociación.
- **Reducción de transacciones:** Una transacción que no contiene frecuentes k -itemsets es inútil en exploraciones posteriores y, por lo tanto, puede ignorarse.
- **Recuento de conjuntos de elementos basado en hash:** Si el recuento de cubos de hash correspondiente de un k -el conjunto de elementos está por debajo de un cierto umbral, k -El conjunto de elementos no puede ser frecuente.
- **Recuento dinámico de conjuntos de elementos:** Solo agregue nuevos conjuntos de elementos candidatos cuando se estima que todos sus subconjuntos son frecuentes.

Resumen

Como técnica de análisis no supervisada que descubre relaciones entre elementos, las reglas de asociación encuentran muchos usos en actividades, incluido el análisis de la cesta de la compra, el análisis del flujo de clics y los motores de recomendación. Aunque las reglas de asociación no se utilizan para predecir resultados o comportamientos, son buenas para identificar relaciones "interesantes" dentro de elementos de un gran conjunto de datos. Muy a menudo, las relaciones reveladas que sugieren las reglas de asociación no parecen obvias; por lo tanto, brindan información valiosa para que las instituciones mejoren sus operaciones comerciales.

El algoritmo Apriori es uno de los primeros y más fundamentales algoritmos para las reglas de asociación. Este capítulo utilizó un ejemplo de una tienda de comestibles para recorrer los pasos de Apriori y generar K -conjuntos de elementos y reglas útiles para el análisis y la visualización posteriores. Se discutieron algunas medidas como el apoyo, la confianza, la elevación y el apalancamiento. Estas medidas juntas ayudan a identificar las reglas interesantes y eliminan las reglas coincidentes. Finalmente, el capítulo discutió algunos pros y contras del algoritmo Apriori y destacó algunos métodos para mejorar su eficiencia.

Ejercicios

1. ¿Qué es la propiedad Apriori?

2. A continuación se muestra una lista de cinco transacciones que incluyen los elementos A, B, C y D:

- T1: { A B C }
- T2: { A, C }
- T3: { ANTES DE CRISTO }
- T4: { A, D }
- T5: { A, C, D }

¿Qué conjuntos de elementos satisfacen el soporte mínimo de 0,5? (Sugerencia: un conjunto de elementos puede incluir más de un elemento).

3. ¿Cómo se identifican las reglas interesantes? ¿Cómo se distinguen las reglas interesantes de las reglas coincidentes?

4. Un minorista local tiene una base de datos que almacena 10,000 transacciones del verano pasado. Después de analizar los datos, un equipo de ciencia de datos ha identificado las siguientes estadísticas:

- {batería} aparece en 6.000 transacciones.
- {protector solar} aparece en 5.000 transacciones.
- {sandalias} aparece en 4.000 transacciones.
- {bochitas} aparece en 2000 transacciones.
- {batería, protector solar} aparece en 1.500 transacciones.
- {batería, sandalias} aparece en 1000 transacciones.
- {batería, cuencos} aparece en 250 transacciones.
- {batería, protector solar, sandalias} aparece en 600 transacciones.

Responde las siguientes preguntas:

a. ¿Cuáles son los valores de soporte de los conjuntos de elementos anteriores?

segundo. Suponiendo que el soporte mínimo es 0,05, ¿qué conjuntos de elementos se consideran frecuentes?

C. ¿Cuáles son los valores de confianza de { batería } → { protector solar } y { batería, protector solar } → { sandalias }? ¿Cuál de las dos reglas es más interesante?

re. Enumere todas las reglas candidatas que se pueden formar a partir de las estadísticas. ¿Qué reglas se consideran interesantes con la confianza mínima de 0,25? De estas interesantes reglas, ¿qué regla se considera la más útil (es decir, la menos coincidente)?

Bibliografía

- [1] P. Hájek, I. Havel y M. Chytíl, "El método GUHA de determinación automática de hipótesis", *Informática*, vol. 1, no. 4, págs. 293-308, 1966.
- [2] R. Agrawal, T. Imielinski y A. Swami, "Reglas de asociación minera entre conjuntos de elementos en grandes bases de datos", *SIGMOD '93 Actas de la Conferencia Internacional ACM SIGMOD 1993 sobre Gestión de Datos*, págs. 207-216, 1993.
- [3] SRA. Chen, JS Park y P. Yu, "Efficient Data Mining for Path Traversal Patterns", *Transacciones IEEE sobre Conocimiento e Ingeniería de Datos*, vol. 10, no. 2, págs. 209-221, 1998.
- [4] R. Cooley, B. Mobasher y J. Srivastava, "Web Mining: Information and Pattern Discovery on the World Wide Web", *Actas de la 9a Conferencia Internacional IEEE sobre Herramientas con Inteligencia Artificial*, págs. 558-567, 1997.
- [5] R. Agrawal y R. Srikant, "Algoritmos rápidos para reglas de asociación minera en grandes bases de datos", en *Actas de la 20a Conferencia Internacional sobre Bases de Datos Muy Grandes*, San Francisco, CA, Estados Unidos, 1994.
- [6] S. Brin, R. Motwani, JD Ullman y S. Tsur, "Reglas de recuento e implicación de conjuntos de elementos dinámicos para los datos de la cesta de la compra", *SIGMOD*, vol. 26, no. 2, págs. 255-264, 1997.
- [7] G. Piatetsky-Shapiro, "Descubrimiento, análisis y presentación de reglas sólidas", *Descubrimiento del conocimiento en Bases de datos*, págs. 229-248, 1991.
- [8] S. Brin, R. Motwani y C. Silverstein, "Beyond Market Baskets: Generalizing Association Rules to Correlations", *Actas de la Conferencia Conjunta ACM SIGACT-SIGMOD / PODS '97*, vol. 26, no. 2, págs. 265-276, 1997.
- [9] CC Aggarwal y PS Yu, "Un nuevo marco para la generación de conjuntos de elementos", en *Actas del Decimoséptimo Simposio ACM SIGACT-SIGMOD-SIGART sobre principios de sistemas de bases de datos (PODS '98)*, Seattle, Washington, Estados Unidos, 1998.
- [10] M. Hahsler, "Una comparación de las medidas de interés comúnmente utilizadas para las reglas de asociación", 9 de marzo 2011. [En línea]. Disponible: http://michael.hahsler.net/research/association_rules/medidas.html. [Consultado el 4 de marzo de 2014].
- [11] W. Lin, SA Álvarez, y C. Ruiz, "Minería de reglas de asociación de apoyo adaptativo eficiente para sistemas de recomendación", *Minería de datos y descubrimiento de conocimientos*, vol. 6, no. 1, págs. 83-105, 2002.
- [12] B. Mobasher, H. Dai, T. Luo y M. Nakagawa, "Personalización efectiva basada en el descubrimiento de reglas de asociación a partir de datos de uso web", en *ACM*, 2011.
- [13] W. Lin, SA Álvarez y C. Ruiz, "Recomendación colaborativa a través de minería de reglas de asociación adaptativa", en *Actas del Taller internacional sobre minería web para comercio electrónico (WEBKDD)*, Boston, MA, 2000.

6

Teoría analítica avanzada y métodos: Regresión

Conceptos clave

Variable categórica

Regresión lineal

Regresión logística

Mínimos cuadrados ordinarios (OLS)

Curva de característica de funcionamiento del receptor (ROC)

Derechos residuales de autor

En general, el análisis de regresión intenta explicar la influencia que tiene un conjunto de variables sobre el resultado de otra variable de interés. A menudo, la variable de resultado se llama **variable dependiente** porque el resultado depende de las otras variables. Estas variables adicionales a veces se denominan **variables de entrada**

o el **variables independientes**. El análisis de regresión es útil para responder los siguientes tipos de preguntas:

- ¿Cuál es el ingreso esperado de una persona?
- ¿Cuál es la probabilidad de que un solicitante no pague un préstamo?

La regresión lineal es una herramienta útil para responder a la primera pregunta, y la regresión logística es un método popular para abordar la segunda. Este capítulo examina estas dos técnicas de regresión y explica cuándo una técnica es más apropiada que la otra.

El análisis de regresión es una herramienta explicativa útil que puede identificar las variables de entrada que tienen la mayor influencia estadística en el resultado. Con tal conocimiento y percepción, se pueden intentar cambios ambientales para producir valores más favorables de las variables de entrada. Por ejemplo, si se encuentra que el nivel de lectura de los estudiantes de 10 años es un excelente predictor del éxito de los estudiantes en la escuela secundaria y un factor en su asistencia a la universidad, entonces se puede considerar, implementar y poner énfasis adicional en la lectura, evaluado para mejorar los niveles de lectura de los estudiantes a una edad más temprana.

6.1 Regresión lineal

La regresión lineal es una técnica analítica que se utiliza para modelar la relación entre varias variables de entrada y una variable de resultado continua. Un supuesto clave es que la relación entre una variable de entrada y la variable de resultado es lineal. Si bien este supuesto puede parecer restrictivo, a menudo es posible transformar adecuadamente las variables de entrada o resultado para lograr una relación lineal entre las variables de entrada y resultado modificadas. Las posibles transformaciones se cubrirán con más detalle más adelante en este capítulo.

Las ciencias físicas tienen modelos lineales bien conocidos, como la Ley de Ohm, que establece que la corriente eléctrica que fluye a través de un circuito resistivo es linealmente proporcional al voltaje aplicado al circuito. Tal modelo se considera determinista en el sentido de que si se conocen los valores de entrada, el valor de la variable de resultado se determina con precisión. Un modelo de regresión lineal es uno probabilístico que da cuenta de la aleatoriedad que puede afectar cualquier resultado en particular. Basado en valores de entrada conocidos, un modelo de regresión lineal proporciona el valor esperado de la variable de resultado en base a los valores de las variables de entrada, pero puede quedar cierta incertidumbre al predecir cualquier resultado en particular. Así, Los modelos de regresión lineal son útiles en aplicaciones de las ciencias físicas y sociales donde puede haber una variación considerable en un resultado particular basado en un conjunto dado de valores de entrada. Después de presentar posibles casos de uso de regresión lineal, se proporcionan los fundamentos del modelado de regresión lineal.

6.1.1 Casos de uso

La regresión lineal se utiliza a menudo en escenarios empresariales, gubernamentales y de otro tipo. Algunas aplicaciones prácticas comunes de la regresión lineal en el mundo real incluyen las siguientes:

- **Bienes raíces:** Se puede utilizar un análisis de regresión lineal simple para calcular los precios de las viviendas residenciales en función de la superficie habitable de la vivienda. Suchamodel ayuda a establecer o evaluar el precio de lista de una persona en el mercado. El modelo podría mejorarse aún más si se incluyen otras variables de entrada, como el número de baños, el número de dormitorios, el tamaño del lote, las clasificaciones del distrito escolar, las estadísticas de delitos y los impuestos sobre la propiedad.

- Previsión de la demanda:** Las empresas y los gobiernos pueden utilizar modelos de regresión lineal para predecir la demanda de bienes y servicios. Por ejemplo, las cadenas de restaurantes pueden prepararse adecuadamente para el tipo y la cantidad previstos de alimentos que los clientes consumirán en función del clima, el día de la semana, si un artículo se ofrece como especial, la hora del día y el volumen de reservas. Se pueden construir modelos similares para predecir las ventas minoristas, las visitas a la sala de emergencias y los envíos de ambulancias.
- Médico:** Se puede utilizar un modelo de regresión lineal para analizar el efecto de un tratamiento de radiación propuesto en la reducción del tamaño de los tumores. Las variables de entrada pueden incluir la duración de un único tratamiento de radiación, la frecuencia del tratamiento de radiación y los atributos del paciente, como la edad o el peso.

6.1.2 Descripción del modelo

Como sugiere el nombre de esta técnica, el modelo de regresión lineal asume que existe una relación lineal entre las variables de entrada y la variable de resultado. Esta relación se puede expresar como se muestra en la Ecuación 6-1.

$$y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_{p-1} X_{p-1} + \epsilon$$

(6-1)

dónde:

y es la variable de resultado

X_j son las variables de entrada, para $j = 1, 2, \dots, p - 1$

β_0 es el valor de y cuando cada X_j es igual a cero

β_j es el cambio en y basado en un cambio de unidad en X_j , para $j = 1, 2, \dots, p - 1$

ϵ es un término de error aleatorio que representa la diferencia en el modelo lineal y un valor observado particular para y

Suponga que se desea construir un modelo de regresión lineal que estime el ingreso anual de una persona como una función de dos variables — edad y educación — ambas expresadas en años. En este caso, el ingreso es la variable de resultado y las variables de entrada son la edad y la educación. Aunque puede ser una generalización excesiva, tal modelo parece intuitivamente correcto en el sentido de que los ingresos de las personas deberían aumentar a medida que su conjunto de habilidades y experiencia se amplíen con la edad. Además, se esperaría que las oportunidades de empleo y los salarios iniciales fueran mayores para aquellos que han obtenido más educación.

Sin embargo, también es obvio que existe una variación considerable en los niveles de ingresos para un grupo de personas con edades y años de educación idénticos. Esta variación está representada por ϵ en el modelo. Entonces, en este ejemplo, el modelo se expresaría como se muestra en la Ecuación 6-2.

$$\text{Ingresos} = \beta_0 + \beta_1 \text{Edad} + \beta_2 \text{Educación} + \epsilon$$

(6-2)

En el modelo lineal, el β_j s representan los p parámetros desconocidos. Las estimaciones de estos parámetros desconocidos se eligen de modo que, en promedio, el modelo proporcione una estimación razonable de los ingresos de una persona.

basado en la edad y la educación. En otras palabras, el modelo ajustado debe minimizar el error general entre el modelo lineal y las observaciones reales. Mínimos cuadrados ordinarios (MCO) es una técnica común para estimar los parámetros.

Para ilustrar cómo funciona MCO, suponga que solo hay una variable de entrada, x , para una variable de resultado y . Además, **norte** observaciones de (x, y) se obtienen y se grafican en la Figura 6-1.

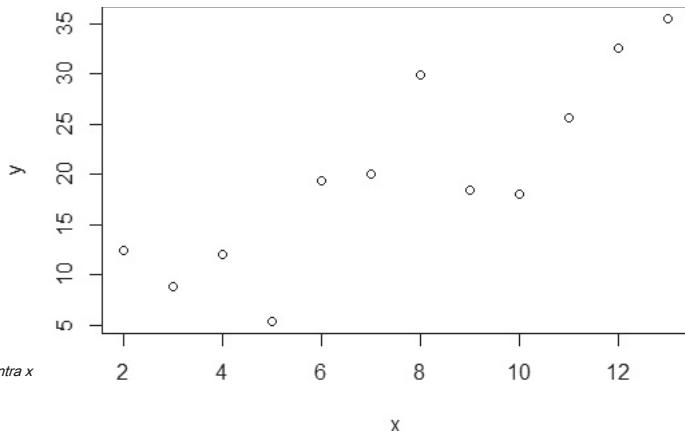


FIGURE 6-1 Diagrama de dispersión de y contra x

El objetivo es encontrar la línea que mejor se aproxime a la relación entre la variable de resultado y las variables de entrada. Con OLS, el objetivo es encontrar la línea que pasa por estos puntos que minimice la suma de los cuadrados de la diferencia entre cada punto y la línea en la dirección vertical.

ción. En otras palabras, encuentre los valores de β_0 y β_1 de manera que se minimice la suma que se muestra en la Ecuación 6-3.

$$\sum_{j=1}^{n_{\text{puntos}}} [y_{j,0} - (\beta_0 + \beta_1 x_{j,0})]^2 \quad (6-3)$$

Las n líneas distan individuales
representan la distan

vertical

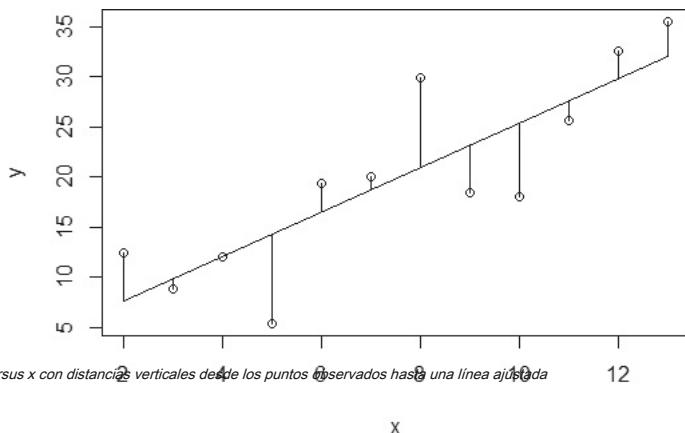


FIGURE 6-2 Diagrama de dispersión de y versus x con distancias verticales de los puntos observados hasta una línea ajustada

En la Figura 3-7 del Capítulo 3, "Revisión de métodos analíticos de datos básicos usando R", el ejemplo del Cuarteto de Anscombe usó OLS para ajustar la línea de regresión lineal a cada uno de los cuatro conjuntos de datos. MCO para múltiples variables de entrada es una extensión directa del caso de una variable de entrada proporcionado en la Ecuación 6-3.

La discusión anterior proporcionó el enfoque para encontrar el mejor ajuste lineal para un conjunto de observaciones. Sin embargo, al hacer algunas suposiciones adicionales sobre el término de error, es posible proporcionar más capacidades al utilizar el modelo de regresión lineal. En general, estos supuestos casi siempre se hacen, por lo que el siguiente modelo, construido sobre el modelo descrito anteriormente, se llama simplemente modelo de regresión lineal.

Modelo de regresión lineal (con errores normalmente distribuidos)

In the previous model description, there were no assumptions made about the error term; no additional assumptions were necessary for OLS to provide estimates of the model parameters. However, in most linear regression analyses, it is common to assume that the error term is a normally distributed random variable with mean equal to zero and constant variance. Thus, the linear regression model is expressed as shown in Equation 6-4.

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_{p-1} x_{p-1} + \varepsilon \quad (6-4)$$

where:

y is the outcome variable

x_j are the input variables, for $j = 1, 2, \dots, p - 1$

β_0 is the value of y when each x_j equals zero

β_j is the change in y based on a unit change in x_j , for $j = 1, 2, \dots, p - 1$

$\varepsilon \sim N(0, \sigma^2)$ and the ε s are independent of each other

This additional assumption yields the following result about the expected value of y , $E(y)$ for given (x_1, x_2, \dots, x_{p-1}):

$$\begin{aligned} E(y) &= E(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_{p-1} x_{p-1} + \varepsilon) \\ &= \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_{p-1} x_{p-1} + E(\varepsilon) \\ &= \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_{p-1} x_{p-1} \end{aligned}$$

Because β_j and x_j are constants, the $E(y)$ is the value of the linear regression model for the given (x_1, x_2, \dots, x_{p-1}). Furthermore, the variance of y , $V(y)$, for given (x_1, x_2, \dots, x_{p-1}) is this:

$$\begin{aligned} V(y) &= V(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_{p-1} x_{p-1} + \varepsilon) \\ &= 0 + V(\varepsilon) = \sigma^2 \end{aligned}$$

Thus, for a given (x_1, x_2, \dots, x_{p-1}), y is normally distributed with mean $\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_{p-1} x_{p-1}$ and variance σ^2 . For a regression model with just one input variable, Figure 6-3 illustrates the normality assumption on the error terms and the effect on the outcome variable, y , for a given value of x .

For $x = 8$, one would expect to observe a value of y near 20, but a value of y from 15 to 25 would appear possible based on the illustrated normal distribution. Thus, the regression model estimates the expected value of y for the given value of x . Additionally, the normality assumption on the error term provides some useful properties that can be utilized in performing hypothesis testing on the linear regression model and

providing confidence in
these statistical techniq

on of
ome.

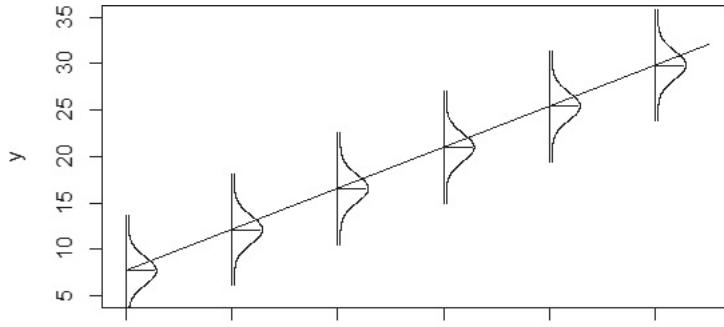


FIGURE 6-3 Normal distribution about y for a given value of x

Example in R

Returning to the **Income** example, in addition to the variables age and education, the person's gender, female or male, is considered an input variable. The following code reads a comma-separated-value (CSV) file of 1,500 people's incomes, ages, years of education, and gender. The first 10 rows are displayed:

```
income_input = as.data.frame( read.csv("c:/data/income.csv") income_input[1:10,] )
```

ID	Income	Age	Education	Gender
1	113 69	12	1	
2	91 52	18	0	
3	121 65	14	0	
4	81 58	12	0	
5	68 31	16	1	
6	92 51	15	1	
7	75 53	15	0	
8	76 56	13	0	
9	56 42	15	1	
10	53 33	11	1	

Each person in the sample has been assigned an identification number, **ID**. **Income** is expressed in thousands of dollars. (For example, 113 denotes \$113,000.) As described earlier, **Age** and **Education** are expressed in years. For **Gender**, a 0 denotes female and a 1 denotes male. A summary of the imported data reveals that the incomes vary from \$14,000 to \$134,000. The ages are between 18 and 70 years. The education experience for each person varies from a minimum of 10 years to a maximum of 20 years.

```
summary(income_input)
```

ID	Income	Age	Education
Min.	1.0	14.00	10.00

1st Qu.: 375.8	1st Qu.: 62.00	1st Qu.: 30.00	1st Qu.: 12.00
Median : 750.5	Median : 76.00	Median : 44.00	Median : 15.00
Mean : 750.5	Mean : 75.99	Mean : 43.58	Mean : 14.68
3rd Qu.: 1125.2	3rd Qu.: 91.00	3rd Qu.: 57.00	3rd Qu.: 16.00
Max. : 1500.0	Max. : 134.00	Max. : 70.00	Max. : 20.00

Gender

Min. : 0.00
1st Qu.: 0.00
Median : 0.00
Mean : 0.49
3rd Qu.: 1.00
Max. : 1.00

As described in Chapter 3, a scatterplot matrix is an informative tool to view the pair-wise relationships of the variables. The basic assumption of a linear regression model is that there is a linear relationship between the outcome variable and the predictor variables.

matrix in Figure 6-4

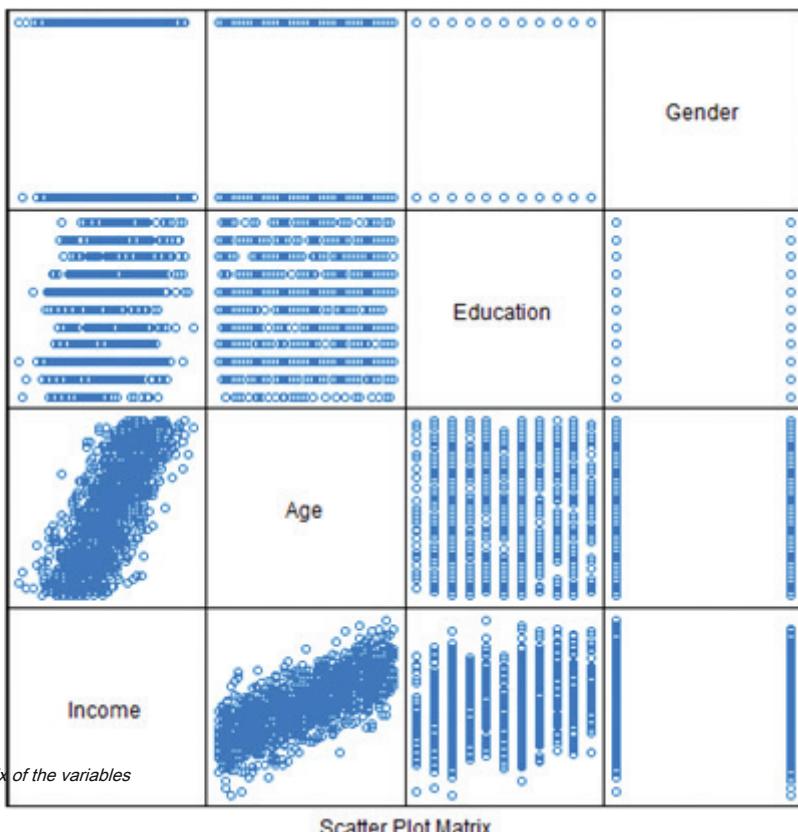


FIGURE 6-4 Scatterplot matrix of the variables

```
library(lattice)

splom(~income_input[c(2:5)], groups=NULL, data=income_input,
      axis.line.tck = 0,
      axis.text.alpha = 0)
```

Because the dependent variable is typically plotted along the y-axis, examine the set of scatterplots along the bottom of thematrix. A strong positive linear trend is observed for **Income** as a function of **Age**.

Against **Education**, a slight positive trend may exist, but the trend is not quite as obvious as is the case with the **Age** variable.

Lastly, there is no observed effect on **Income** based on **Gender**.

With this qualitative understanding of the relationships between **Income** and the input variables, it seems reasonable to quantitatively evaluate the linear relationships of these variables. Utilizing the normality assumption applied to the error term, the proposed linear regression model is shown in Equation 6-5.

$$\text{Income} = \beta_0 + \beta_1 \text{Age} + \beta_2 \text{Education} + \beta_3 \text{Gender} + \varepsilon \quad (6-5)$$

Using the linear model function, lm(), in R, the income model can be applied to the data as follows:

```
results <- lm(income ~ Age + Education + Gender, income_input) summary(results)
```

Call:

```
lm(formula = Income ~ Age + Education + Gender, data = income_input)
```

Residuals:

	Min	1Q	Median	3Q	Max
	-37.340	-8.101	0.139	7.885	37.271

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	7.26299	1.95575	3.714	0.000212 ***
Age	0.99520	0.02057	48.373	< 2e-16 ***
Education	1.75788	0.11581	15.179	< 2e-16 ***
Gender	-0.93433	0.62388	-1.498	0.134443

Signif. code	s: 0 ****	0.001 **	* 0.01	** 0.05 * 0.1 ** 1

Residual standard error: 12.07 on 1496 degrees of freedom Multiple R-squared: 0.6364,

Adjusted R-squared: 0.6357 F-statistic:

873 on 3 and 1496 DF, p-value: < 2.2e-16

The intercept term, β_0 , is implicitly included in the model. The lm() function performs the parameter estimation for the parameters β_j ($j = 0, 1, 2, 3$) using ordinary least squares and provides several useful calculations and results that are stored in the variable called **results** in this example.

After the stated call to lm(), a few statistics on the residuals are displayed in the output. The residuals are the observed values of the error term for each of the n observations and are defined for $i = 1, 2, \dots, n$, as shown in Equation 6-6.

$$\epsilon_i = y_i - (\beta_0 + \beta_1 x_{i,1} + \beta_2 x_{i,2} + \dots + \beta_{p-1} x_{i,p-1}) \quad (6-6)$$

where b_j denotes the estimate for parameter β_j for $j = 0, 1, 2, \dots, p-1$