

Manuseio / Manipulação de dados no R

Marcus Suassuna Santos

10/08/2020

Estrutura do Curso

- Recapitular leitura de dados
- Seleções (*subset*) de dados
- Transformações de tabelas
- Usando pacote `dplyr`

Leitura de dados

Exemplo: dados hidrológicos do hidroweb

Visualização preliminar dos dados

vazoes_T_15400000.txt - Bloco de Notas

Arquivo Editar Formatar Exibir Ajuda

Sistema de Informações Hidrológicas

Versão Web 3.0

© 2018 Agência Nacional de Águas (ANA)

NivelConsistencia: 1 = Bruto, 2 = Consistido

MediaDiaria: 0 = Não, 1 = Sim

MetodoObtencaoVazoes: 1 = Curva de descarga, 2 = Transferência de vazões, 3 = Soma de vazões, 4 = ADCP

Status: 0 = Branco, 1 = Real, 2 = Estimado, 3 = Duvidoso, 4 = Régua Seca

Restrições da consulta:

Código da Estação:15400000

```
EstacaoCodigo;NivelConsistencia;Data;Hora;MediaDiaria;MetodoObtencaoVazoes;Maxima;Minima;Media;DiaMaxima;DiaMinima;MaximaStatus;MinimaStatus;MediaStatus;MediaAnual;MediaAnualStatus;
15400000;1;01/05/2020;1;1;24928,809;16828,121;21510,004;4;31;1;1;1;0;24218,432;24495,201;24510,623;24928,809;24495,201;24325,881;23669,533;23684,697;23806,178;23066,912;22457,164;
15400000;1;01/04/2020;1;1;35765,723;24249,107;31806,418;5;30;1;1;1;0;33714,969;34659,691;35265,895;35580,195;35765,723;35376,668;34916,238;34550,031;34131,242;33229,504;32747,51;
15400000;1;01/03/2020;1;1;38220,746;29217,846;33417,98;1;20;1;1;1;0;38220,746;37971,637;38029,051;38029,051;37818,734;37304,938;36869,984;36400,059;35710,016;34604,84;34058,664;3;
15400000;1;01/02/2020;1;1;37837,832;26668,504;33767,098;29;3;3;3;0;26700,67;26829,521;26668,504;26845,648;27104,344;28966,131;30630,035;31828,986;33014,855;33480,793;34222,07;3;
15400000;1;01/01/2020;1;1;32552,133;27169,209;29749,174;13;31;1;1;1;0;27347,98;27609,041;28366,391;29133,822;30664,469;30475,32;31009,85;31479,133;31706,32;32180,742;32286,641;32;
15400000;1;01/12/2019;1;1;27055,746;9523,957;18665,463;30;1;1;1;1;0;9523,957;9543,34;10969,012;12053,295;12536,544;12140,463;11772,134;12359,735;12983,972;13301,764;14030,618;151;
15400000;1;01/11/2019;1;1;12669,962;5245,145;8123,22;21;9;1;1;1;0;6762,743;6241,065;5922,611;5945,634;5634,79;5508,313;5281,316;5310,342;5245,145;5361,325;5419,886;5552,791;5500,;
15400000;1;01/10/2019;1;1;7493,213;3689,381;5429,159;14;3;1;1;1;0;4094,575;3701,539;3689,381;3701,539;3824,198;3781,043;3911,229;3824,198;3762,622;3774,898;4730,974;6415,326;7211,;
15400000;1;01/09/2019;1;1;1;0;0;0;0;4655,759;4113,777;4043,584;3948,824;3923,741;3799,508;3824,198;3811,843;3604,829;3811,843;3707,625;3683,309;3683,309;3738,129;3725,913;
15400000;1;01/08/2019;1;1;1;0;0;0;0;8305,734;8570,383;8533,637;8351,072;8515,293;8206,419;8009,553;5405,216;5464,011;5361,325;5339,446;5144,514;5108,807;4813,702;
15400000;1;01/07/2019;1;1;1;0;0;0;0;16281,389;15398,104;15044,301;14947,412;14670,574;14408,155;13948,689;13508,056;15178,025;14574,88;13948,689;13233,347;12492,227;12118,643;
15400000;1;01/06/2019;1;1;27331,705;17033,824;22977,828;1;30;1;1;1;0;27331,705;27023,371;25965,697;25334,797;24742,52;24866,637;24356,623;24495,201;25366,162;24866,637;25053,385;2;
15400000;1;01/05/2019;1;1;33968,051;25902,264;31066,391;5;29;1;1;1;0;33696,926;33733,016;33841,395;33787,184;33968,051;33841,395;33301,207;32907,789;32410,404;32057,41;32357,334;3;
15400000;1;01/04/2019;1;1;41832,559;32587,613;37698,945;4;27;1;1;1;0;41412,23;41532,109;41612,125;41832,559;41532,109;41013,859;40934,41;40498,809;40577,84;40914,563;40993,988;411;
15400000;1;01/03/2019;1;1;44067,676;40736,125;42384,902;8;24;1;1;1;0;42093,809;42864,273;43519,48;43656,344;43628,961;43738,52;43793,332;44067,676;43793,332;43903,008;43738,52;436;
15400000;1;01/02/2019;1;1;41973,129;33877,559;39156,477;25;2;1;1;1;0;34167,559;33877,559;35008,086;35339,723;36400,059;37058,785;38086,512;38624,852;38837,359;39089,246;40163,781;
15400000;1;01/01/2019;1;1;33552,762;28899,189;31725,742;23;7;1;1;1;0;31881,629;30871,469;30184,135;30013,492;30218,32;29234,666;28899,189;28899,189;29707,533;30149,969;30630,035;3;
15400000;1;01/12/2018;1;1;30836,922;22501,508;26396,133;28;1;1;1;1;0;22501,508;22887,617;24187,773;24680,574;23973,713;24341,248;26220,193;26700,67;27478,359;27347,98;27120,553;26;
15400000;1;01/11/2018;1;1;22236,094;9504,595;13220,458;30;8;1;1;1;0;10700,672;11114,856;11114,856;10782,89;10233,933;10073,989;9836,4;9504,595;9582,162;9504,595;9994,483;9994,483;
15400000;1;01/10/2018;1;1;19254,642;3948,824;5502,076;31;8;1;1;1;0;4427,03;4655,759;4374,036;4216,938;4242,925;4453,646;3973,985;3948,824;4308,235;4139,45;4011,874;4255,947;4387,25;
15400000;1;01/09/2018;1;1;6241,065;3961,395;4923,697;1;24;1;1;1;0;6241,065;6069,167;6092,469;6030,429;5747,561;5582,54;5288,565;5044,843;5073,222;5030,683;4862,286;5016,542;4918,1;
15400000;1;01/08/2018;1;1;7493,213;5216,993;6259,088;14;8;1;1;1;0;6371,602;6371,602;6106,821;5911,289;5654,68;5496,694;5294,083;5216,993;5309,556;5402,791;6657,995;7018,919;7405,4;
15400000;1;01/07/2018;1;1;1;0;0;0;0;13587,507;14023,938;13519,118;13111,782;12334,53;12094,312;11964,106;12488,41;12029,137;11427,866;11174,086;10652,958;10304,479;9840,649;9;
15400000;1;01/06/2018;1;1;1;0;0;0;0;23922,74;22598,049;22261,262;20935,072;21538,734;22654,383;23534,945;24487,498;24255,145;23021,963;22289,248;21621,602;21483,564;20744,486;
15400000;1;01/05/2018;1;1;30653,346;24400,258;28243,049;1;31;1;1;1;0;30653,346;30430,186;30049,191;30239,441;30080,865;30398,359;29512,846;29324,5;29261,828;28918,127;29512,846;29;
15400000;1;01/04/2018;1;1;1;0;0;0;0;40215,695;39610,145;38702,363;37685,210;37365,668;36903,871;37153,082;38210,566;40324,441;39070,098;38863,601;38210,566;37525,344;37046,018;
```

Leitura de dados

Lendo os dados

```
Cotas <- read.table("dados/cotas_T_14990000.txt", header = TRUE, sep = ";")  
str(Cotas)
```

```
## 'data.frame': 222698 obs. of 1 variable:
```

```
## $ Sistema.de.Informações.Hidrológicas: chr "Versão Web 3.0" "@ 2018 Agência Nacional de Águas (
```

Leitura de dados

Lendo os dados

```
Cotas <- read.csv2("dados/cotas_T_14990000.txt",  
                 row.names = NULL,  
                 skip = 12)  
colnames(Cotas) <- c(colnames(Cotas)[-1], "x")  
head(Cotas, 2)
```

```
## EstacaoCodigo NivelConsistencia      Data      Hora MediaDiaria  
## 1      14990000      1 01/06/2014      1  
## 2      14990000      1 01/06/2014 01/01/1900 07:00:00      0  
## TipoMedicaoCotas Maxima Minima Media DiaMaxima DiaMinima MaximaStatus  
## 1      1      2948      2935      2945      12      1      0  
## 2      1      2948      2935      2945      12      1      0  
## MinimaStatus MediaStatus MediaAnual MediaAnualStatus Cota01 Cota02 Cota03  
## 1      0      0      NA      0      2936      2940      2941  
## 2      0      0      NA      0      2935      2939      2940  
## Cota04 Cota05 Cota06 Cota07 Cota08 Cota09 Cota10 Cota11 Cota12 Cota13 Cota14  
## 1      2943      2944      2944      2944      2945      2946      2946      2947      2948      2948      2948  
## 2      2942      2944      2944      2944      2944      2945      2946      2946      2947      2948      2948  
## Cota15 Cota16 Cota17 Cota18 Cota19 Cota20 Cota21 Cota22 Cota23 Cota24 Cota25  
## 1      2948      2948      2946      2945      2946      2946      2946      2946      2946      2945      2946
```

Leitura de dados

Lendo os dados

```
library(readr)
Cotas <- read_csv2("dados/cotas_T_14990000.txt",
                  skip = 12)
head(Cotas)
```

```
## # A tibble: 6 x 78
##   EstacaoCodigo NivelConsistenc~ Data Hora MediaDiaria TipoMedicaoCotas Maxima
##   <dbl> <dbl> <chr> <chr> <dbl> <dbl> <dbl>
## 1 14990000 1 01/0~ <NA> 1 1 2948
## 2 14990000 1 01/0~ 01/0~ 0 1 2948
## 3 14990000 1 01/0~ 01/0~ 0 1 2948
## 4 14990000 1 01/0~ 01/0~ 0 1 2934
## 5 14990000 1 01/0~ 01/0~ 0 1 2934
## 6 14990000 1 01/0~ <NA> 1 1 2934
## # ... with 71 more variables: Minima <dbl>, Media <dbl>, DiaMaxima <dbl>,
## # DiaMinima <dbl>, MaximaStatus <dbl>, MinimaStatus <dbl>, MediaStatus <dbl>,
## # MediaAnual <dbl>, MediaAnualStatus <dbl>, Cota01 <dbl>, Cota02 <dbl>,
## # Cota03 <dbl>, Cota04 <dbl>, Cota05 <dbl>, Cota06 <dbl>, Cota07 <dbl>,
## # Cota08 <dbl>, Cota09 <dbl>, Cota10 <dbl>, Cota11 <dbl>, Cota12 <dbl>,
## # Cota13 <dbl>, Cota14 <dbl>, Cota15 <dbl>, Cota16 <dbl>, Cota17 <dbl>,
```

Leitura de dados

Ajustando os dados: convertendo Datas

```
library(dplyr)
Cotas <- as.data.frame(Cotas)
# Mais adiante entro em detalhes
Cotas <- Cotas %>%
  filter(complete.cases(Maxima, Media, Minima))
str(Cotas)

## 'data.frame':   2849 obs. of  78 variables:
## $ EstacaoCodigo      : num  1.5e+07 1.5e+07 1.5e+07 1.5e+07 1.5e+07 ...
## $ NivelConsistencia: num  1 1 1 1 1 1 1 1 1 1 ...
## $ Data                : chr  "01/06/2014" "01/06/2014" "01/06/2014" "01/05/2014" ...
## $ Hora                : chr  NA "01/01/1900 07:00:00" "01/01/1900 17:00:00" "01/01/1900 07:00:00" ...
## $ MediaDiaria        : num  1 0 0 0 0 1 1 0 0 0 ...
## $ TipoMedicaoCotas    : num  1 1 1 1 1 1 1 1 1 1 ...
## $ Maxima             : num  2948 2948 2948 2934 2934 ...
## $ Minima             : num  2935 2935 2935 2801 2801 ...
## $ Media              : num  2945 2945 2945 2874 2874 ...
## $ DiaMaxima          : num  12 12 12 31 31 31 30 30 30 31 ...
## $ DiaMinima          : num  1 1 1 1 1 1 1 1 1 1 ...
## $ MaximaStatus       : num  0 0 0 0 0 0 0 0 0 0 ...
```

Leitura de dados

Ajustando os dados: convertendo Datas

```
Cotas$Data <- as.Date(Cotas$Data, format = "%d/%m/%Y")
str(Cotas)
```

```
## 'data.frame': 2849 obs. of 78 variables:
## $ EstacaoCodigo : num 1.5e+07 1.5e+07 1.5e+07 1.5e+07 1.5e+07 ...
## $ NivelConsistencia: num 1 1 1 1 1 1 1 1 1 1 ...
## $ Data : Date, format: "2014-06-01" "2014-06-01" ...
## $ Hora : chr NA "01/01/1900 07:00:00" "01/01/1900 17:00:00" "01/01/1900 07:00:00" ...
## $ MediaDiaria : num 1 0 0 0 0 1 1 0 0 0 ...
## $ TipoMedicaoCotas : num 1 1 1 1 1 1 1 1 1 1 ...
## $ Maxima : num 2948 2948 2948 2934 2934 ...
## $ Minima : num 2935 2935 2935 2801 2801 ...
## $ Media : num 2945 2945 2945 2874 2874 ...
## $ DiaMaxima : num 12 12 12 31 31 31 30 30 30 31 ...
## $ DiaMinima : num 1 1 1 1 1 1 1 1 1 1 ...
## $ MaximaStatus : num 0 0 0 0 0 0 0 0 0 0 ...
## $ MinimaStatus : num 0 0 0 0 0 0 0 0 0 0 ...
## $ MediaStatus : num 0 0 0 0 0 0 0 0 0 0 ...
## $ MediaAnual : num NA NA NA NA NA NA NA NA NA NA ...
## $ MediaAnualStatus : num 0 0 0 0 0 0 0 0 0 0 ...
```


Explorando os dados

```
dim(Cotas)
```

```
## [1] 2849 78
```

```
names(Cotas)
```

```
## [1] "EstacaoCodigo" "NivelConsistencia" "Data"  
## [4] "Hora" "MediaDiaria" "TipoMedicaoCotas"  
## [7] "Maxima" "Minima" "Media"  
## [10] "DiaMaxima" "DiaMinima" "MaximaStatus"  
## [13] "MinimaStatus" "MediaStatus" "MediaAnual"  
## [16] "MediaAnualStatus" "Cota01" "Cota02"  
## [19] "Cota03" "Cota04" "Cota05"  
## [22] "Cota06" "Cota07" "Cota08"  
## [25] "Cota09" "Cota10" "Cota11"  
## [28] "Cota12" "Cota13" "Cota14"  
## [31] "Cota15" "Cota16" "Cota17"  
## [34] "Cota18" "Cota19" "Cota20"  
## [37] "Cota21" "Cota22" "Cota23"  
## [40] "Cota24" "Cota25" "Cota26"  
## [43] "Cota27" "Cota28" "Cota29"  
## [46] "Cota30" "Cota31" "Cota01Status"
```

Explorando os dados

```
str(Cotas)
```

```
## 'data.frame':    2849 obs. of  78 variables:
## $ EstacaoCodigo      : num  1.5e+07 1.5e+07 1.5e+07 1.5e+07 1.5e+07 ...
## $ NivelConsistencia: num  1 1 1 1 1 1 1 1 1 1 ...
## $ Data               : Date, format: "2014-06-01" "2014-06-01" ...
## $ Hora               : chr  NA "01/01/1900 07:00:00" "01/01/1900 17:00:00" "01/01/1900 07:00:00" ...
## $ MediaDiaria        : num  1 0 0 0 0 1 1 0 0 0 ...
## $ TipoMedicaoCotas    : num  1 1 1 1 1 1 1 1 1 1 ...
## $ Maxima             : num  2948 2948 2948 2934 2934 ...
## $ Minima             : num  2935 2935 2935 2801 2801 ...
## $ Media              : num  2945 2945 2945 2874 2874 ...
## $ DiaMaxima          : num  12 12 12 31 31 31 30 30 30 31 ...
## $ DiaMinima          : num  1 1 1 1 1 1 1 1 1 1 ...
## $ MaximaStatus       : num  0 0 0 0 0 0 0 0 0 0 ...
## $ MinimaStatus       : num  0 0 0 0 0 0 0 0 0 0 ...
## $ MediaStatus        : num  0 0 0 0 0 0 0 0 0 0 ...
## $ MediaAnual         : num  NA NA NA NA NA NA NA NA NA NA ...
## $ MediaAnualStatus   : num  0 0 0 0 0 0 0 0 0 0 ...
## $ Cota01             : num  2936 2935 2937 2801 2805 ...
## $ Cota02             : num  2940 2939 2940 2808 2811 ...
## $ Cota03             : num  2941 2940 2941 2814 2817 ...
## $ Cota04             : num  2943 2942 2943 2819 2822 ...
```

Explorando os dados

```
table(sapply(Cotas, class))
```

```
##  
## character      Date      numeric  
##           1           1           76
```

Selecionando dados

A forma mais direta de selecionar partes de tabelas (linhas e colunas) é usando os colchetes.

Exemplo, selecionando a 10a linha:

```
Cotas[10,]
```

```
## EstacaoCodigo NivelConsistencia Data Hora MediaDiaria
## 10 14990000 1 2014-03-01 01/01/1900 07:00:00 0
## TipoMedicaoCotas Maxima Minima Media DiaMaxima DiaMinima MaximaStatus
## 10 1 2665 2469 2564 31 1 0
## MinimaStatus MediaStatus MediaAnual MediaAnualStatus Cota01 Cota02 Cota03
## 10 0 0 NA 0 2469 2474 2479
## Cota04 Cota05 Cota06 Cota07 Cota08 Cota09 Cota10 Cota11 Cota12 Cota13 Cota14
## 10 2484 2489 2496 2502 2508 2515 2522 2531 2539 2544 2549
## Cota15 Cota16 Cota17 Cota18 Cota19 Cota20 Cota21 Cota22 Cota23 Cota24 Cota25
## 10 2555 2562 2569 2572 2577 2585 2591 2598 2605 2613 2621
## Cota26 Cota27 Cota28 Cota29 Cota30 Cota31 Cota01Status Cota02Status
## 10 2630 2636 2643 2650 2656 2662 NA NA
## Cota03Status Cota04Status Cota05Status Cota06Status Cota07Status
## 10 NA NA NA NA NA
## Cota08Status Cota09Status Cota10Status Cota11Status Cota12Status
```

Selecionando dados

Exemplo, selecionando a terceira coluna:

```
Cotas[,3]
```

```
##      [1] "2014-06-01" "2014-06-01" "2014-06-01" "2014-05-01" "2014-05-01"  
##      [6] "2014-05-01" "2014-04-01" "2014-04-01" "2014-04-01" "2014-03-01"  
##     [11] "2014-03-01" "2014-03-01" "2014-02-01" "2014-02-01" "2014-02-01"  
##     [16] "2014-01-01" "2014-01-01" "2014-01-01" "2013-12-01" "2013-12-01"  
##     [21] "2013-12-01" "2013-11-01" "2013-11-01" "2013-11-01" "2013-10-01"  
##     [26] "2013-10-01" "2013-10-01" "2013-09-01" "2013-09-01" "2013-09-01"  
##     [31] "2013-08-01" "2013-08-01" "2013-08-01" "2013-07-01" "2013-07-01"  
##     [36] "2013-07-01" "2013-06-01" "2013-06-01" "2013-06-01" "2013-05-01"  
##     [41] "2013-05-01" "2013-05-01" "2013-04-01" "2013-04-01" "2013-04-01"  
##     [46] "2013-03-01" "2013-03-01" "2013-03-01" "2013-02-01" "2013-02-01"  
##     [51] "2013-02-01" "2013-01-01" "2013-01-01" "2013-01-01" "2012-12-01"  
##     [56] "2012-12-01" "2012-12-01" "2012-11-01" "2012-11-01" "2012-11-01"  
##     [61] "2012-10-01" "2012-10-01" "2012-10-01" "2012-09-01" "2012-09-01"  
##     [66] "2012-09-01" "2012-08-01" "2012-08-01" "2012-08-01" "2012-07-01"  
##     [71] "2012-07-01" "2012-07-01" "2012-06-01" "2012-06-01" "2012-06-01"  
##     [76] "2012-05-01" "2012-05-01" "2012-05-01" "2012-04-01" "2012-04-01"  
##     [81] "2012-04-01" "2012-03-01" "2012-03-01" "2012-03-01" "2012-02-01"  
##     [86] "2012-02-01" "2012-02-01" "2012-01-01" "2012-01-01" "2012-01-01"
```

Selecionando dados

Exemplo, selecionando mais de uma coluna:

```
Cotas[,2:3]
```

```
##      NivelConsistencia      Data
## 1                1 2014-06-01
## 2                1 2014-06-01
## 3                1 2014-06-01
## 4                1 2014-05-01
## 5                1 2014-05-01
## 6                1 2014-05-01
## 7                1 2014-04-01
## 8                1 2014-04-01
## 9                1 2014-04-01
## 10               1 2014-03-01
## 11               1 2014-03-01
## 12               1 2014-03-01
## 13               1 2014-02-01
## 14               1 2014-02-01
## 15               1 2014-02-01
## 16               1 2014-01-01
## 17               1 2014-01-01
```

Selecionando dados

Exemplo, selecionando usando o nome da coluna:

```
Cotas[, "Media"]
```

```
##      [1] 2945 2945 2945 2874 2874 2874 2741 2741 2741 2564 2564 2564 2397 2397
##      [15] 2397 2276 2276 2276 2162 2162 2162 1982 1982 1982 2090 2090 2090 2460
##      [29] 2460 2460 2691 2691 2691 2860 2860 2860 2927 2927 2927 2867 2867 2867
##      [43] 2740 2740 2740 2605 2605 2605 2338 2338 2338 2082 2082 2082 1751 1751
##      [57] 1751 1660 1660 1660 1741 1741 1741 2197 2197 2197 2675 2675 2675 2866
##      [71] 2866 2866 2964 2964 2964 2972 2972 2972 2847 2847 2847 2702 2702 2702
##      [85] 2517 2517 2517 2233 2233 2233 1923 1923 1923 1758 1758 1758 1723 1723
##      [99] 1723 2082 2082 2082 2627 2627 2627 2822 2822 2822 2854 2854 2854 2756
##     [113] 2756 2756 2579 2579 2579 2401 2401 2401 2179 2179 2179 1954 1954 1954
##     [127] 1771 1771 1771 1479 1479 1479 1492 1492 1492 1913 1913 1913 2460 2460
##     [141] 2460 2707 2707 2707 2787 2787 2787 2737 2737 2737 2529 2529 2529 2357
##     [155] 2357 2357 2250 2250 2250 2115 2115 2115 1761 1761 1761 1716 1716 1716
##     [169] 1915 1915 1915 2469 2469 2471 2797 2797 2797 2949 2949 2949 2946 2946
##     [183] 2946 2900 2900 2900 2815 2815 2815 2680 2680 2680 2563 2563 2563 2353
##     [197] 2353 2353 2105 1906 1895 2204 2604 2798 2850 2763 2670 2576 2437 2275
##     [211] 2045 1860 1827 2177 2608 2778 2806 2708 2523 2361 2345 2235 2048 1766
##     [225] 1751 2132 2545 2782 2874 2803 2680 2619 2477 2238 1980 1710 1543 1861
##     [239] 2344 2662 2789 2779 2653 2493 2246 2151 2029 1939 2036 2291 2531 2651
```

Selecionando dados

Exemplo, selecionando mais de uma coluna, utilizando os nomes:

```
Cotas[,c("Data", "Media", "Maxima", "Cota01")]
```

##	Data	Media	Maxima	Cota01
## 1	2014-06-01	2945	2948	2936
## 2	2014-06-01	2945	2948	2935
## 3	2014-06-01	2945	2948	2937
## 4	2014-05-01	2874	2934	2801
## 5	2014-05-01	2874	2934	2805
## 6	2014-05-01	2874	2934	2803
## 7	2014-04-01	2741	2798	2671
## 8	2014-04-01	2741	2798	2668
## 9	2014-04-01	2741	2798	2673
## 10	2014-03-01	2564	2665	2469
## 11	2014-03-01	2564	2665	2472
## 12	2014-03-01	2564	2665	2471
## 13	2014-02-01	2397	2467	2336
## 14	2014-02-01	2397	2467	2335
## 15	2014-02-01	2397	2467	2337
## 16	2014-01-01	2276	2334	2228
## 17	2014-01-01	2276	2334	2224

Selecionando dados

Utilizando a função `subset`:

```
subset(Cotas, select = c(Data, Cota01:Cota31))
```

##	Data	Cota01	Cota02	Cota03	Cota04	Cota05	Cota06	Cota07	Cota08	Cota09
## 1	2014-06-01	2936	2940	2941	2943	2944	2944	2944	2945	2946
## 2	2014-06-01	2935	2939	2940	2942	2944	2944	2944	2944	2945
## 3	2014-06-01	2937	2940	2941	2943	2944	2944	2944	2945	2946
## 4	2014-05-01	2801	2808	2814	2819	2824	2829	2833	2839	2843
## 5	2014-05-01	2805	2811	2817	2822	2827	2831	2836	2841	2846
## 6	2014-05-01	2803	2810	2816	2821	2826	2830	2835	2840	2845
## 7	2014-04-01	2671	2679	2686	2691	2695	2698	2702	2710	2716
## 8	2014-04-01	2668	2677	2684	2690	2694	2697	2700	2708	2714
## 9	2014-04-01	2673	2681	2687	2692	2696	2699	2704	2711	2718
## 10	2014-03-01	2469	2474	2479	2484	2489	2496	2502	2508	2515
## 11	2014-03-01	2472	2477	2482	2487	2493	2499	2505	2512	2519
## 12	2014-03-01	2471	2476	2481	2486	2491	2498	2504	2510	2517
## 13	2014-02-01	2336	2339	2343	2347	2352	2357	2361	2365	2369
## 14	2014-02-01	2335	2338	2341	2346	2350	2355	2360	2364	2368
## 15	2014-02-01	2337	2340	2344	2348	2353	2358	2362	2366	2370
## 16	2014-01-01	2228	2234	2239	2242	2244	2246	2248	2249	2251
## 17	2014-01-01	2224	2231	2237	2240	2243	2245	2247	2248	2250

Selecionando dados

Também é bastante comum utiliza o operador `$` para selecionar colunas. O resultado é um vetor do mesmo tipo da coluna do data frame original.

Cotas\$Media

```
##      [1] 2945 2945 2945 2874 2874 2874 2741 2741 2741 2564 2564 2564 2397 2397
##     [15] 2397 2276 2276 2276 2162 2162 2162 1982 1982 1982 2090 2090 2090 2460
##     [29] 2460 2460 2691 2691 2691 2860 2860 2860 2927 2927 2927 2867 2867 2867
##     [43] 2740 2740 2740 2605 2605 2605 2338 2338 2338 2082 2082 2082 1751 1751
##     [57] 1751 1660 1660 1660 1741 1741 1741 2197 2197 2197 2675 2675 2675 2866
##     [71] 2866 2866 2964 2964 2964 2972 2972 2972 2847 2847 2847 2702 2702 2702
##     [85] 2517 2517 2517 2233 2233 2233 1923 1923 1923 1758 1758 1758 1723 1723
##     [99] 1723 2082 2082 2082 2627 2627 2627 2822 2822 2822 2854 2854 2854 2756
##    [113] 2756 2756 2579 2579 2579 2401 2401 2401 2179 2179 2179 1954 1954 1954
##    [127] 1771 1771 1771 1479 1479 1479 1492 1492 1492 1913 1913 1913 2460 2460
##    [141] 2460 2707 2707 2707 2787 2787 2787 2737 2737 2737 2529 2529 2529 2357
##    [155] 2357 2357 2250 2250 2250 2115 2115 2115 1761 1761 1761 1716 1716 1716
##    [169] 1915 1915 1915 2469 2469 2471 2797 2797 2797 2949 2949 2949 2946 2946
##    [183] 2946 2900 2900 2900 2815 2815 2815 2680 2680 2680 2563 2563 2563 2353
##    [197] 2353 2353 2105 1906 1895 2204 2604 2798 2850 2763 2670 2576 2437 2275
##    [211] 2045 1860 1827 2177 2608 2778 2806 2708 2523 2361 2345 2235 2048 1766
##    [225] 1751 2132 2545 2782 2874 2803 2680 2619 2477 2238 1980 1710 1543 1861
```

Selecionando dados

Aplicar filtros baseados em valores das variáveis.

```
head(Cotas, 2)
```

```
## EstacaoCodigo NivelConsistencia      Data      Hora MediaDiaria
## 1      14990000      1 2014-06-01      <NA>      1
## 2      14990000      1 2014-06-01 01/01/1900 07:00:00      0
## TipoMedicaoCotas Maxima Minima Media DiaMaxima DiaMinima MaximaStatus
## 1      1      2948      2935      2945      12      1      0
## 2      1      2948      2935      2945      12      1      0
## MinimaStatus MediaStatus MediaAnual MediaAnualStatus Cota01 Cota02 Cota03
## 1      0      0      NA      0      2936      2940      2941
## 2      0      0      NA      0      2935      2939      2940
## Cota04 Cota05 Cota06 Cota07 Cota08 Cota09 Cota10 Cota11 Cota12 Cota13 Cota14
## 1      2943      2944      2944      2944      2945      2946      2946      2947      2948      2948      2948
## 2      2942      2944      2944      2944      2944      2945      2946      2946      2947      2948      2948
## Cota15 Cota16 Cota17 Cota18 Cota19 Cota20 Cota21 Cota22 Cota23 Cota24 Cota25
## 1      2948      2948      2946      2945      2946      2946      2946      2946      2946      2945      2946
## 2      2948      2948      2946      2945      2945      2946      2946      2946      2946      2945      2945
## Cota26 Cota27 Cota28 Cota29 Cota30 Cota31 Cota01Status Cota02Status
## 1      2946      2947      2947      2948      2948      NA      NA      NA
## 2      2945      2946      2947      2947      2947      NA      NA      NA
```

Selecionando dados

Aplicar filtros baseados em valores das variáveis.

```
Cotas <- Cotas[Cotas$NivelConsistencia == 1,]  
head(Cotas, 2)
```

```
## EstacaoCodigo NivelConsistencia      Data      Hora MediaDiaria  
## 1      14990000      1 2014-06-01      <NA>      1  
## 2      14990000      1 2014-06-01 01/01/1900 07:00:00      0  
## TipoMedicaoCotas Maxima Minima Media DiaMaxima DiaMinima MaximaStatus  
## 1      1      2948      2935      2945      12      1      0  
## 2      1      2948      2935      2945      12      1      0  
## MinimaStatus MediaStatus MediaAnual MediaAnualStatus Cota01 Cota02 Cota03  
## 1      0      0      NA      0      2936      2940      2941  
## 2      0      0      NA      0      2935      2939      2940  
## Cota04 Cota05 Cota06 Cota07 Cota08 Cota09 Cota10 Cota11 Cota12 Cota13 Cota14  
## 1      2943      2944      2944      2944      2945      2946      2946      2947      2948      2948      2948  
## 2      2942      2944      2944      2944      2944      2945      2946      2946      2947      2948      2948  
## Cota15 Cota16 Cota17 Cota18 Cota19 Cota20 Cota21 Cota22 Cota23 Cota24 Cota25  
## 1      2948      2948      2946      2945      2946      2946      2946      2946      2946      2945      2946  
## 2      2948      2948      2946      2945      2945      2946      2946      2946      2946      2945      2945  
## Cota26 Cota27 Cota28 Cota29 Cota30 Cota31 Cota01Status Cota02Status  
## 1      2946      2947      2947      2948      2948      NA      NA      NA
```

Selecionando dados

Aplicar filtros baseados em valores, utilizando mais de uma condição: operador & (e):

```
Cotas <- Cotas[Cotas$NivelConsistencia == 1 &  
              Cotas$MediaDiaria == 1,]
```

```
head(Cotas, 2)
```

```
## EstacaoCodigo NivelConsistencia      Data Hora MediaDiaria TipoMedicaoCotas  
## 1      14990000      1 2014-06-01 <NA>      1      1  
## 6      14990000      1 2014-05-01 <NA>      1      1  
## Maxima Minima Media DiaMaxima DiaMinima MaximaStatus MinimaStatus MediaStatus  
## 1   2948   2935  2945      12      1      0      0      0  
## 6   2934   2801  2874      31      1      0      0      0  
## MediaAnual MediaAnualStatus Cota01 Cota02 Cota03 Cota04 Cota05 Cota06 Cota07  
## 1      NA      0   2936   2940   2941   2943   2944   2944   2944  
## 6      NA      0   2803   2810   2816   2821   2826   2830   2835  
## Cota08 Cota09 Cota10 Cota11 Cota12 Cota13 Cota14 Cota15 Cota16 Cota17 Cota18  
## 1   2945   2946   2946   2947   2948   2948   2948   2948   2948   2946   2945  
## 6   2840   2845   2850   2856   2861   2865   2869   2872   2877   2881   2885  
## Cota19 Cota20 Cota21 Cota22 Cota23 Cota24 Cota25 Cota26 Cota27 Cota28 Cota29  
## 1   2946   2946   2946   2946   2946   2945   2946   2946   2947   2947   2948  
## 6   2889   2893   2897   2900   2905   2910   2915   2920   2922   2925   2927
```

Selecionando dados

Aplicar filtros baseados em valores.

```
Cotas$Data > as.Date("2014-01-01")
```

```
## [1] TRUE TRUE TRUE TRUE TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [13] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [25] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [37] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [49] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [61] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [73] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [85] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [97] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [109] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [121] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [133] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [145] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [157] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [169] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [181] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [193] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
## [205] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
```

Selecionando dados

Aplicar filtros baseados em valores.

```
Cotas[Cotas$Data > as.Date("2014-01-01"), ]
```

```
##      EstacaoCodigo NivelConsistencia      Data Hora MediaDiaria TipoMedicaoCotas
## 1      14990000      1 2014-06-01 <NA>          1          1
## 6      14990000      1 2014-05-01 <NA>          1          1
## 7      14990000      1 2014-04-01 <NA>          1          1
## 12     14990000      1 2014-03-01 <NA>          1          1
## 13     14990000      1 2014-02-01 <NA>          1          1
##      Maxima Minima Media DiaMaxima DiaMinima MaximaStatus MinimaStatus
## 1      2948  2935  2945      12          1          0          0
## 6      2934  2801  2874      31          1          0          0
## 7      2798  2668  2741      30          1          0          0
## 12     2665  2469  2564      31          1          0          0
## 13     2467  2335  2397      28          1          0          0
##      MediaStatus MediaAnual MediaAnualStatus Cota01 Cota02 Cota03 Cota04 Cota05
## 1          0      NA          0  2936  2940  2941  2943  2944
## 6          0      NA          0  2803  2810  2816  2821  2826
## 7          0      NA          0  2671  2679  2686  2691  2695
## 12         0      NA          0  2471  2476  2481  2486  2491
## 13         0      NA          0  2336  2339  2343  2347  2352
```

Seleccionando datos

Aplicar filtros basados en valores.

```
dim(Cotas[Cotas$Data > as.Date("2014-01-01"), ])
```

```
## [1] 5 78
```


Seleccionando dados

Aplicar filtros baseados em valores.

```
summary(Cotas$Maxima)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      1580   2163   2467   2421   2713   2997
```

```
quantile(Cotas$Maxima,
         probs = c(0.5, 0.80, 0.9, 0.98), na.rm = TRUE)
```

```
##      50%      80%      90%      98%
## 2467.0 2754.0 2818.0 2910.2
```

Selecionando dados

Aplicar filtros baseados em valores.

```
Cotas[Cotas$Maxima > quantile(Cotas$Maxima,  
                               probs = c(0.98),  
                               na.rm = TRUE), ]
```

##	EstacaoCodigo	NivelConsistencia	Data	Hora	MediaDiaria
## 1	14990000	1	2014-06-01	<NA>	1
## 6	14990000	1	2014-05-01	<NA>	1
## 37	14990000	1	2013-06-01	<NA>	1
## 42	14990000	1	2013-05-01	<NA>	1
## 70	14990000	1	2012-07-01	<NA>	1
## 75	14990000	1	2012-06-01	<NA>	1
## 76	14990000	1	2012-05-01	<NA>	1
## 81	14990000	1	2012-04-01	<NA>	1
## 180	14990000	1	2009-07-01	<NA>	1
## 183	14990000	1	2009-06-01	<NA>	1
## 186	14990000	1	2009-05-01	<NA>	1
## 312	14990000	1	1999-07-01	<NA>	1
## 313	14990000	1	1999-06-01	<NA>	1
## 464	14990000	1	1989-07-01	<NA>	1
## 465	14990000	1	1989-06-01	<NA>	1

Selecionando dados

Aplicar filtros baseados em valores.

```
head(Cotas[Cotas$Maxima > quantile(Cotas$Maxima,  
                                   probs = c(0.98),  
                                   na.rm = TRUE), ],
```

2)

```
## EstacaoCodigo NivelConsistencia      Data Hora MediaDiaria TipoMedicaoCotas  
## 1      14990000      1 2014-06-01 <NA>      1      1  
## 6      14990000      1 2014-05-01 <NA>      1      1  
## Maxima Minima Media DiaMaxima DiaMinima MaximaStatus MinimaStatus MediaStatus  
## 1   2948   2935  2945      12      1      0      0      0  
## 6   2934   2801  2874      31      1      0      0      0  
## MediaAnual MediaAnualStatus Cota01 Cota02 Cota03 Cota04 Cota05 Cota06 Cota07  
## 1      NA      0   2936   2940   2941   2943   2944   2944   2944  
## 6      NA      0   2803   2810   2816   2821   2826   2830   2835  
## Cota08 Cota09 Cota10 Cota11 Cota12 Cota13 Cota14 Cota15 Cota16 Cota17 Cota18  
## 1   2945   2946   2946   2947   2948   2948   2948   2948   2946   2945  
## 6   2840   2845   2850   2856   2861   2865   2869   2872   2877   2881   2885  
## Cota19 Cota20 Cota21 Cota22 Cota23 Cota24 Cota25 Cota26 Cota27 Cota28 Cota29  
## 1   2946   2946   2946   2946   2946   2945   2946   2946   2947   2947   2948  
## 6   2889   2893   2897   2900   2905   2910   2915   2920   2922   2925   2927
```

Selecionando dados

Aplicar filtros baseados em valores.

```
Cotas[Cotas$Maxima >
  quantile(Cotas$Maxima,
    probs = c(0.98),
    na.rm = TRUE), ]$Data

## [1] "2014-06-01" "2014-05-01" "2013-06-01" "2013-05-01" "2012-07-01"
## [6] "2012-06-01" "2012-05-01" "2012-04-01" "2009-07-01" "2009-06-01"
## [11] "2009-05-01" "1999-07-01" "1999-06-01" "1989-07-01" "1989-06-01"
## [16] "1976-07-01" "1976-06-01" "1976-05-01" "1975-06-01" "1971-06-01"
## [21] "1953-07-01" "1953-06-01" "1953-05-01" "1922-07-01" "1922-06-01"
## [26] "1922-05-01" "1909-06-01"
```

Selecionando dados

Aplicar filtros baseados em valores.

```
Selecao <- Cotas[Cotas$Maxima >
              quantile(Cotas$Maxima,
                        probs = c(0.98),
                        na.rm = TRUE), ]$Data
table(format(Selecao, "%Y"))
```

```
##
## 1909 1922 1953 1971 1975 1976 1989 1999 2009 2012 2013 2014
##   1    3    3    1    1    3    2    2    3    4    2    2
```

```
table(format(Selecao, "%b"))
```

```
##
## abr jul jun mai
##   1   7  12   7
```

Selecionando dados

Aplicar filtros baseados em valores, utilizando mais de uma condição:
operador & (e):

```
library(lubridate)
Cotas[Cotas$Maxima >
      quantile(Cotas$Maxima,
              probs = c(0.98),
              na.rm = TRUE) &
      year(Cotas$Data) != 2014, ]
```

##	EstacaoCodigo	NivelConsistencia	Data	Hora	MediaDiaria
## 37	14990000	1	2013-06-01	<NA>	1
## 42	14990000	1	2013-05-01	<NA>	1
## 70	14990000	1	2012-07-01	<NA>	1
## 75	14990000	1	2012-06-01	<NA>	1
## 76	14990000	1	2012-05-01	<NA>	1
## 81	14990000	1	2012-04-01	<NA>	1
## 180	14990000	1	2009-07-01	<NA>	1
## 183	14990000	1	2009-06-01	<NA>	1
## 186	14990000	1	2009-05-01	<NA>	1
## 312	14990000	1	1999-07-01	<NA>	1
## 313	14990000	1	1999-06-01	<NA>	1

Selecionando dados

Aplicar filtros baseados em valores, utilizando mais de uma condição:
operador | (ou):

```
Cotas[Cotas$Maxima == max(Cotas$Maxima) |  
      Cotas$Minima == min(Cotas$Minima), ]
```

```
##      EstacaoCodigo NivelConsistencia      Data Hora MediaDiaria  
## 75      14990000      1 2012-06-01 <NA>      1  
## 76      14990000      1 2012-05-01 <NA>      1  
## 133     14990000      1 2010-10-01 <NA>      1  
##      TipoMedicaoCotas Maxima Minima Media DiaMaxima DiaMinima MaximaStatus  
## 75      1 2997 2923 2964      1      30      0  
## 76      1 2997 2925 2972      28      1      0  
## 133     1 1672 1363 1492      1      24      0  
##      MinimaStatus MediaStatus MediaAnual MediaAnualStatus Cota01 Cota02 Cota03  
## 75      0      0      NA      0 2997 2996 2995  
## 76      0      0      NA      0 2927 2931 2935  
## 133     0      0      NA      0 1670 1659 1647  
##      Cota04 Cota05 Cota06 Cota07 Cota08 Cota09 Cota10 Cota11 Cota12 Cota13  
## 75      2994 2992 2989 2987 2985 2983 2982 2980 2976 2974  
## 76      2939 2943 2947 2950 2954 2958 2962 2965 2967 2970  
## 133     1635 1621 1609 1600 1589 1580 1572 1564 1554 1542
```

Selecionando dados

Utilizando a função `subset`:

```
subset(Cotas,  
       subset = (Cotas$Maxima > quantile(Cotas$Maxima, probs = c(0.98),  
                                         na.rm = TRUE) &  
                year(Cotas$Data) == 2014),  
       select = c(Data, Maxima, Media))
```

```
##           Data Maxima Media  
## 1 2014-06-01   2948  2945  
## 6 2014-05-01   2934  2874
```


Transformando dados

Eventualmente deseja-se criar dados adicionais e adicioná-los à tabela original. Primeiramente, vamos fazer uma seleção das colunas dos dados originais.

```
Minimas <- Cotas[, c("Minima")]  
Cotas <- Cotas[, c("Data", "Maxima")]  
head(Cotas)
```

```
##           Data Maxima  
## 1  2014-06-01   2948  
## 6  2014-05-01   2934  
## 7  2014-04-01   2798  
## 12 2014-03-01   2665  
## 13 2014-02-01   2467  
## 18 2014-01-01   2334
```

Transformando dados

Agora, vamos supor que eu queira o dado normalizado, ou seja, subtrair da média e dividir pelo desvio padrão, adicionando este dado em forma de uma coluna adicional à tabela original. Para fazer isso, basta executar nominar a coluna e adicioná-la atribuindo a ela valores, usando o operador <-

```
Cotas$Normalizada <- scale(Cotas$Maxima)
head(Cotas)
```

```
##           Data Maxima Normalizada
## 1 2014-06-01   2948   1.5881566
## 6 2014-05-01   2934   1.5459523
## 7 2014-04-01   2798   1.1359680
## 12 2014-03-01  2665   0.7350274
## 13 2014-02-01  2467   0.1381384
## 18 2014-01-01  2334  -0.2628022
```

Transformando dados

Para remover a coluna adicionada, ou qualquer outra, basta atribuir o valor **NULL** à coluna que se deseja remover.

```
Cotas$Normalizada <- NULL  
head(Cotas)
```

```
##           Data Maxima  
## 1  2014-06-01   2948  
## 6  2014-05-01   2934  
## 7  2014-04-01   2798  
## 12 2014-03-01   2665  
## 13 2014-02-01   2467  
## 18 2014-01-01   2334
```

Transformando dados

Com linhas, não funciona da mesma forma.

```
Cotas2 <- Cotas  
Cotas2 <- Cotas2[-1, ]  
head(Cotas2)
```

```
##           Data Maxima  
## 6  2014-05-01   2934  
## 7  2014-04-01   2798  
## 12 2014-03-01   2665  
## 13 2014-02-01   2467  
## 18 2014-01-01   2334  
## 19 2013-12-01   2221
```

Transformando dados

Outra forma de adicionar dados a data frames é utilizando as funções `rbind` e `cbind`. Por exemplo, suponha que desejo acrescentar manualmente os dados de 2015. Supondo que o nível máximo que o rio alcançou em 2015 tenha sido o nível de 2700 cm em Junho.

```
Cotas2015 <- data.frame("Data" = as.Date("2015-06-01"),  
                      "Maxima" = 2700)
```

```
Cotas2015
```

```
##           Data Maxima  
## 1 2015-06-01  2700
```

Transformando dados

Agora os dados serão unidos por linhas ("rows"). Assim será usada a função `rbind()` ("row" + "bind").

```
Cotas <- rbind(Cotas2015, Cotas)
head(Cotas)
```

```
##           Data Maxima
## 1  2015-06-01   2800
## 11 2014-06-01   2948
## 6   2014-05-01   2934
## 7   2014-04-01   2798
## 12  2014-03-01   2665
## 13  2014-02-01   2467
```

Transformando dados

Suponha, agora que eu queira incluir novamente as mínimas. Agora posso usar a função `cbind()` (“column” + “bind”).

```
Cotas <- Cotas[-1,]  
Cotas <- cbind(Cotas, Minimas)  
head(Cotas,10)
```

```
##           Data Maxima Minimas  
## 11 2014-06-01   2948   2935  
##  6 2014-05-01   2934   2801  
##  7 2014-04-01   2798   2668  
## 12 2014-03-01   2665   2469  
## 13 2014-02-01   2467   2335  
## 18 2014-01-01   2334   2224  
## 19 2013-12-01   2221   2070  
## 24 2013-11-01   2068   1935  
## 25 2013-10-01   2298   1940  
## 30 2013-09-01   2576   2304
```

Renomeando variáveis

É possível consultar e alterar os nomes das colunas.

```
names(Cotas)
```

```
## [1] "Data" "Maxima" "Minimas"
```

```
names(Cotas)[2:3] <- c("MaximaMensal", "MinimaMensal")
```

```
head(Cotas)
```

```
##           Data MaximaMensal MinimaMensal
## 11 2014-06-01          2948           2935
## 6  2014-05-01          2934           2801
## 7  2014-04-01          2798           2668
## 12 2014-03-01          2665           2469
## 13 2014-02-01          2467           2335
## 18 2014-01-01          2334           2224
```


Pacote `dplyr`

Todas as funções do R básico são também executáveis usando o pacote `dplyr`. Mas porque escolheria o pacote ao invés das funções base do R?

[Four reasons why you should check out the R package dplyr.](#)

- Velocidade.
- Simplicidade de conexão, melhorando o manuseio de grandes bancos de dados
- Encadeamento das funções, sem criação de objetos intermediários
- Facilidade da sintaxe e uso.

Execução de transformações usando o pacote `dplyr`

```
library(dplyr)
```

Vamos ler o arquivo de cotas de Manaus.

```
Cotas <- read_csv2("dados/cotas_T_14990000.txt", skip = 12)
head(Cotas)
```

```
## # A tibble: 6 x 78
##   EstacaoCodigo NivelConsistenc~ Data Hora MediaDiaria TipoMedicaoCotas Maxima
##   <dbl> <dbl> <chr> <chr> <dbl> <dbl> <dbl>
## 1 14990000 1 01/0~ <NA> 1 1 2948
## 2 14990000 1 01/0~ 01/0~ 0 1 2948
## 3 14990000 1 01/0~ 01/0~ 0 1 2948
## 4 14990000 1 01/0~ 01/0~ 0 1 2934
## 5 14990000 1 01/0~ 01/0~ 0 1 2934
## 6 14990000 1 01/0~ <NA> 1 1 2934
## # ... with 71 more variables: Minima <dbl>, Media <dbl>, DiaMaxima <dbl>,
## # DiaMinima <dbl>, MaximaStatus <dbl>, MinimaStatus <dbl>, MediaStatus <dbl>,
## # MediaAnual <dbl>, MediaAnualStatus <dbl>, Cota01 <dbl>, Cota02 <dbl>,
## # Cota03 <dbl>, Cota04 <dbl>, Cota05 <dbl>, Cota06 <dbl>, Cota07 <dbl>,
```

Execução de transformações usando o pacote

dp1yr

O pacote `dp1yr` inclui o operador `%>%` (“pipe”), bastante útil para encadear funções. O operador significa que o resultado da função anterior ao “pipe” será o primeiro argumento da função posterior a ela. Vejamos:

```
Cotas <- read_csv2("dados/cotas_T_14990000.txt", skip = 12) %>%
  filter(complete.cases(Maxima, Media, Minima))
head(Cotas)
```

```
## # A tibble: 6 x 78
##   EstacaoCodigo NivelConsistenc~ Data Hora MediaDiaria TipoMedicaoCotas Maxima
##   <dbl> <dbl> <chr> <chr> <dbl> <dbl> <dbl>
## 1 14990000 1 01/0~ <NA> 1 1 2948
## 2 14990000 1 01/0~ 01/0~ 0 1 2948
## 3 14990000 1 01/0~ 01/0~ 0 1 2948
## 4 14990000 1 01/0~ 01/0~ 0 1 2934
## 5 14990000 1 01/0~ 01/0~ 0 1 2934
## 6 14990000 1 01/0~ <NA> 1 1 2934
## # ... with 71 more variables: Minima <dbl>, Media <dbl>, DiaMaxima <dbl>,
## # DiaMinima <dbl>, MaximaStatus <dbl>, MinimaStatus <dbl>, MediaStatus <dbl>,
## # MediaAnual <dbl>, MediaAnualStatus <dbl>, Cota01 <dbl>, Cota02 <dbl>,
```

Execução de transformações usando o pacote `dplyr`

Vamos ver como funciona a função `filter` do pacote `dplyr`. Digite `?filter` no R. O primeiro argumento da função `filter` é uma série `x`. Mas na chamada da função acima, ele não deve ser usado, pois o operador `%>%` indica que a saída da função

```
read_csv2("dados/cotas_T_14990000.txt", skip = 12)
```

será o primeiro argumento da função.

Esse operador que permite o encadeamento simples de funções usando o pacote `dplyr`.

Execução de transformações usando o pacote

dp1yr

```
Cotas <- read_csv2("dados/cotas_T_14990000.txt", skip = 12) %>%
  filter(complete.cases(Maxima, Media, Minima)) %>%
  filter(NivelConsistencia == 1 & Cotas$MediaDiaria == 1) %>%
  select(c("Data", "Maxima")) %>%
  mutate(Data = as.Date(Data, "%d/%m/%Y"),
         Normalizada = scale(Maxima)) %>%
  arrange(Data)
```

```
head(Cotas)
```

```
## # A tibble: 6 x 3
##   Data          Maxima Normalizada[,1]
##   <date>        <dbl>         <dbl>
## 1 1902-10-01    1970          -1.36
## 2 1902-11-01    1774          -1.95
## 3 1902-12-01    1909          -1.54
## 4 1903-01-01    2094          -0.986
## 5 1903-02-01    2172          -0.751
## 6 1903-03-01    2332          -0.269
```

Execução de transformações usando o pacote `dp1yr`

```
Cotas %>%  
  arrange(Maxima) %>%  
  head()
```

```
## # A tibble: 6 x 3  
##   Data          Maxima Normalizada[,1]  
##   <date>      <dbl>      <dbl>  
## 1 1963-10-01   1580      -2.54  
## 2 1916-10-01   1582      -2.53  
## 3 1906-10-01   1589      -2.51  
## 4 1926-10-01   1593      -2.50  
## 5 1906-11-01   1606      -2.46  
## 6 1995-10-01   1625      -2.40
```

Execução de transformações usando o pacote `dp1yr`

```
Cotas %>%
```

```
  arrange(desc(Maxima)) %>%
```

```
  head()
```

```
## # A tibble: 6 x 3
```

```
##   Data          Maxima Normalizada[,1]
```

```
##   <date>        <dbl>         <dbl>
```

```
## 1 2012-05-01    2997            1.74
```

```
## 2 2012-06-01    2997            1.74
```

```
## 3 2009-07-01    2977            1.68
```

```
## 4 2009-06-01    2976            1.67
```

```
## 5 1953-06-01    2970            1.65
```

```
## 6 1953-05-01    2962            1.63
```

Execução de transformações usando o pacote **dplyr**

[Reference dplyr.](#)