





Albert Einstein College of Medicine

## **R** intensive

Session 1 May 14<sup>th</sup>, 2024





#### Welcome!

Question: How many people have experience with coding or analysis in some software package?

- EXCEL
- SPSS
- STATA
- SAS
- PYTHON
- MATLAB
- R

## The goals of this workshop

- This is an introductory workshop!
- We want you to walk away today feeling comfortable:
  - 1. navigating the four panels of Rstudio
  - 2. importing data into R
  - 3. performing some data manipulations and cleaning using dplyr
  - 4. summarizing your data
  - 5. plotting your data

#### Like any language, you need to practice a lot to get fluent!

## Why R?

- Cleaning, analyzing, plotting all in one place
- **Reproducibility** you can save your code and know exactly what you did. You can also rerun code if you get new data
- It's freely available

## A few things to know first

- R has <u>many</u> contributed packages (>10,000) to extend R for basic and advanced analysis
- Typically, a **package** will include code, documentation for the package, the functions inside, and data sets.
- An example of a package is the *gam* package. This package contains multiple functions for fitting generalized additive models. Another example is the *shiny* package, which make interactive, web apps with R.
- Packages are stored in libraries

## The R console

Serves as the primary interface for executing R code. The R console serves as the primary interface for executing R code.

#### 🖻 💾 🖻 🖻 🗘 🥮 🎒

R Console	
R version 4.3.3 (2024-02-29 ucrt) "Angel Food Cake"	
Copyright (C) 2024 The R Foundation for Statistical Computing	
Platform: x86_64-w64-mingw32/x64 (64-bit)	
R is free software and comes with ABSOLUTELY NO WARRANTY.	
You are welcome to redistribute it under certain conditions.	
Type 'license()' or 'licence()' for distribution details.	
Natural language support but running in an English locale	
R is a collaborative project with many contributors.	
Type 'contributors()' for more information and	
'citation()' on how to cite R or R packages in publications.	
Type 'demo()' for some demos, 'help()' for on-line help, or	
'help.start()' for an HTML browser interface to help.	
Type 'q()' to quit R.	
[Previously saved workspace restored]	
4	

## R Studio

• **R Studio** provides a user-friendly interface with various tools and features such as debugging and fill-ins.

8 RStudio					- 0 ×
<u>File Edit Code View Plots Session Build Debug Profile Tools Help</u>					
💿 🔹 🧐 🛫 🗸 🔒 🔚 🦾 🥻 Addins 👻					Project: (None)
Untitled1 × R data sets × P Untitled5 × P Untitled2 × D Untitled3* × D Untitled4 ×			Environment History	Connections Tutorial	
(aux) 🔊 🔒 🕞 Source on Save 🔍 🧨 - 📋	Run 🐤	👌 🖯 🕩 Source 🔹 🖹	💣 📊 📑 Import Datas	set 🔹 🌗 283 MiB 👻 🔏	≡ List • 🛛 🕑 •
1 starwars.cut %>%			R 👻 🦺 Global Environm	ent 👻	Q
2 filter(gender=="feminine") %>%			Data		
4 mutate(height.mean=mean(height.feet.na.rm=TRUE)) %>%			🔘 aa	383 obs. of 33 variables	
<pre>5 filter(height.feet &gt; height.mean,na.rm=TRUE)</pre>			aecom	96 obs. of 34 variables	
6			aecom.compare	481 obs. of 34 variables	
			0 all	2885 obs. of 4 variables	
			0 ana	86 obs. of 35 variables	
			0 anova	List of 13	0
			APO	num [1:385, 1:42] 38 19	41 25 26 38 34
			APO_logit1	Large glm (31 elements,	1.2 MB) Q
6:1 (Top Level) ¢		R Script ¢	APO. all	811 obs. of 31 variables	
		-	APO.final.train	811 obs. of 31 variables	
Console Terminal × Background Jobs ×		= _	APO.final.trai	num [1:385, 1:32] 1.45 -	2.44 2.06 -1.21
Barriss Of 100 50 black vellow blue 40 fema fem	n Mirial 5.4:	5 5.40	APO.final.xtest	num [1:310, 1:30] -0.71	0.775 0.563 1.2
6 Dormé 165 NA brown light brown NA fema fem	in Naboo 5.41	1 5.40	APO.final.xtes	num [1:152, 1:30] -0.71	0.775 0.563 1.2
7 Zam Wesell 168 55 blonde fair, gre yellow NA fema fem	inZolan 5.51	1 5.40	🜔 APO.final.xtra	811 obs. of 30 variables	
9 locasta Nu 167 NA white fair blue NA fema fem	in Coruscant 5.48	9 5.40 8 5.40	APO.imputed.sc	811 obs. of 31 variables	
10 Shaak Ti 178 57 none red, blue black NA fema fem	in Shili 5.84	4 5.40	Files Plots Packages	Help Viewer Presentation	
11 Padmé Amid 165 45 brown light brown 46 fema fem	in Naboo 5.41	1 5.40	🌰 🔿 🏠 🔊		Q
> colnames(starwars)			R: Order rows using col	lumn values - Find in Topic	
[1] "name" "height" "mass" "hair_color" "skin_color" "eye_color	" "birth_year" "sex"				
[9] "gender" "homeworld" "species" "films" "vehicles" "starships	<i>n</i>		arrange {dplyr}		R Documentation
+ filter(gender=="feminine") %>%			<u> </u>		
+ mutate(height.feet=height/30.48) %>%			Order rows u	ising column values	6
+ mutate(height.mean=mean(height.feet,na.rm=TRUE)) %>%					
+ Tilter(neight.reet > neight.mean,na.rm=ikuE) # A tibble: 11 x 12			Description		
name height mass hair_color skin_color eye_color birth_year sex gen	der homeworld height.feet	t height.mean	arrange () orders the	rows of a data frame by the value	s of selected columns
<pre><chr> <int> <db1> <chr> &lt;</chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></chr></db1></int></chr></pre>	r> <chr> <db1></db1></chr>	> <db1></db1>			
2 Avla Secura 178 55 none blue bazel 48 fema fem	in Ryloth 5.84	1 5.40 4 5.40	Unlike other dplyr verbs	s, arrange () largely ignores grou	uping; you need to
3 Adi Gallia 184 50 none dark blue NA fema fem	in Coruscant 6.04	4 5.40	group by them, and fun	ictions of variables are evaluated of	once per data frame, not
4 Luminara U 170 56.2 black yellow blue 58 fema fem	in Mirial 5.58	8 5.40	once per group.		
6 Dormé 165 NA brown light brown NA fema fem	in Naboo 5.43	5 5.40	Usens		
7 Zam Wesell 168 55 blonde fair, gre yellow NA fema fem	in Zolan 5.51	1 5.40	usage		
8 Taun We 213 NA none grey black NA fema fem	inKamino 6.99	9 5.40	arrange(.data,	., .by group = FALSE)	
9 Jocasta Nu 16/ NA white fair blue NA fema fem 10 Shaak Ti 178 57 none red blue black NA fema fem	in Shili 5.48	8 5.40 4 5.40			
11 Padmé Amid 165 45 brown light brown 46 fema fem	in Naboo 5.41	1 5.40	## S3 method for	class 'data.frame'	
>		-	arrange(.data,	., .by_group = FALSE, .lo	cale = NULL)

#### The 4 panels of R Studio

Due but we got service and sub large got got       Important         Important </th <th>R RStudio</th> <th>- 0 X</th> <th></th>	R RStudio	- 0 X	
Image: Section of the section of th	File Edit Code View Plots Session Build Debug Profile Tools Help		
Protect       Total and       Protect	•     • </td <td>Project: (None) 🔻</td> <td></td>	Project: (None) 🔻	
<pre></pre>	● Untitled1 × R data sets × ● Untitled5 × ● Untitled2 × ● Untitled3* × ● Untitled4 ×	Environment History Connections Tutorial	
The standard of the st	A Source on Stee 9 2 - 1	🐨 🗔 🐨 Import Dataset 🔹 🌗 283 Miß 🔹 🖌 🗮 List 🔹 💮 -	
Intercended **emining** 0.5%         Intercended **emining** 0.5% <td< td=""><td>1 starwars.cut %</td><td>R • Global Environment •</td><td></td></td<>	1 starwars.cut %	R • Global Environment •	
A setting of the setting of	2 filter(gender="feminine") %%	Data	
Filter drugter: Feet - height maan, na. mented for	3 mutate(height.reet=height/30.48) %>% 4 mutate(height.mean(height feet na rm-TRUF)) %>%	aa 383 obs of 33 variables	
Image: Compared and the second and	5 filter(height.feet > height.mean.na.rm=TRUE)	acom 96 obs. of 34 variables	
Source panel – write code you can save and reue111	6	Decom compare     481 obs. of 34 variables	
Source panel – write code you can save and ready to the the second seco		All     2885 obs. of 4 variables	
Source panel – write code you can save and reueIII <td></td> <td>Vali     2005 005. 01 4 valiables</td> <td>Environmen</td>		Vali     2005 005. 01 4 valiables	Environmen
(i)       (fg=tem):	/ Source panel – Write code you can save and reuse	and 80 obs. of 55 variables	
at       Togisted 3       Togisted 3       Togisted 3       Togisted 4			variables
Image low 1:       Event for the low 1:       Event f		APO num [1:385, 1:42] 38 19 41 23 20 38 34	
<pre>ter (000 term) = term term term term, term,</pre>		APO_logici Large gim (SI elements, I.2 MB) Q	
Control       Tennini I       Edgeword data         Control       Martinis Vir.       Obs 20       Disk       Vir.       Vir.       Disk       Disk<	611 (lop Level) * R Scriet *	APO.all 811 obs. of 31 variables	
Image: Start of 10 and 11 a	Console Terminal × Background Jobs ×	APO. TINAI. Train 811 ODS. OF 31 Variables	
D darriss 0r. 100 30 black yellow blue 40 rema. remn. Mirial 5.43 5.40 C dorne 165 kb brown fair, gre_yellow MA fema. femin. Naboo 5.41 5.40 C dorne 125 kb brown fair, gre_yellow MA fema. femin. Naboo 5.41 5.40 C dorne 225 Kh orbit 12 Nation 10.755 0.535 1.2. C dorne 225 kb brown light brown 46 fema. femin. Shill 5.44 Filter fema. Section 2 Nation 2 Nati	(R k4.3.3 · ~/ ⇒	APO.final.trai num [1:385, 1:32] 1.45 -2.44 2.06 -1.21	
Contrast 1 128 we provide fraiting the provide of the set of t	Barriss Of., 166 50 black yellow blue 40 fema, femin, Mirial 5.45 5.40	APO. TIMAT. XLEST HUM [1:510, 1:30] $-0.710.7750.5631.2$	
name       prime       prim<	7 Jam Wesell 168 55 blonde fair, gre vellow NA fema femin Zolan 5.51 5.40	APO final xtra $811$ obs. of 30 variables	
0 ocasta Nu 167 NA white fair blue NA fema_femin. Conscant 5.48 5.40 10 fhadmé Amid. 165 45 brown light brown A6 fema_femin. Naboo 5.41 5.40 > contames (starwars) [1] name "height: mass" "hair_color" "skin_color" "eye_color" "birth_year" "sex" 9 Gonames (starwars) [1] name "height: mass" "hair_color skin_color "eye_color" "birth_year" "sex" 9 Gonames (starwars) [1] name "height: mass hair_color skin_color eye_color birth_year sex gender homeworld height. feet height. feat height: feet height. feet h	8 Taun We 213 NA none grey black NA fema femin Kamino 6.99 5.40	APO imputed sc. 811 obs. of 31 variables	
In spanda (11 1/8 37 hone red, blue likk with rema. remin. Shill 3 3.84 5.40         In spanda (mid. 15 37 hone ride, blue black i 1/1 hame: "height: "mass" "hair_color" "skin_color" "eye_color" "birth_year" "sex"         In spanda (mid. 15 36 brown light brown do fema. femin. Shill 5.40         Status: cut & S%         + filter (height: feet.height: 30.48) %%         + mutat (height: feet.height: 30.48) %%         + filter (height: feet.height: 30.48) %%         + make         + filter (height: feet.height: 30.48) %%         + make         + filter (height: feet.height: 30.48) %         - filter (height: feet.h	9 Jocasta Nu 167 NA white fair blue NA fema femin Coruscant 5.48 5.40	AFO. Imputed. Sc OII ODS. OF SI Variables	
Colpares (Starwars) (1) name" "height" "mass" "hair_color" "skin_color" "eye_color" "birth_year" "sex" (9) gender" "nomeworld" "species" "films" "vehicles" "starships" (b) gender" "nomeworld" "species" "films" "vehicles" "starships" (c) names (Starwars) (1) name" "height" "mass" "hair_color" "skin_color" "eye_color" "birth_year" "sex" (c) names (Starwars) (c) name (	10 Sphaak 11 1/8 57 none red, blue black NA Tema Temin Shili 5.84 5.40	Files Plots Packages Help Viewer Presentation	
Col planes (starwars)          [1] Tane"       "height"       "mass"       "hair_color"       "eye_color"       "birth_year"       "sex"         [9] Toender"       "homeworld"       "species"       "fills", "whicles"       "starger, color"       R Documentation         [9] Toender"       "homeworld"       "species"       "fills", "whicles"       "starger, color"       R Documentation         [9] Toender"       "homeworld"       "species"       "fills", "which is the ight.mean.mean (height.feet, na.rm=TRUE))       Swe       R Documentation       Coder rows using column values - End, in topic         [9] Toender"       "homeworld height.feet, height.mean.mean.mean(height.feet, height.mean.mean.mean(height.feet, height.mean.mean.mean(height.feet, height.mean.mean.mean(height.feet, height.mean.mean.mean(height.feet, height.mean.mean.mean(height.feet, height.mean.mean.mean(height.feet, height.mean.mean.mean(height.feet, height.mean.mean.mean.mean(height.feet, height.mean.mean.mean.mean.mean.mean.mean.mean			
11       Trane"       "height"       "mass"       "hair_color"       "secolor"       "birth_year"       "sex"         19       gender"       "nomeworld"       "species"       "time"       "vehicles"       "starships"         19       gender"       "nomeworld"       "species"       "time"       "vehicles"       "starships"         19       gender"       "films"       "vehicles"       "starships"       R Documentation         19       gender       "nomeworld"       "species"       "starships"       R Documentation         11       tible       tible <td>&gt; co names(starwars)</td> <td>R: Order rows using column values - Find in Topic</td> <td></td>	> co names(starwars)	R: Order rows using column values - Find in Topic	
Statistics (ut %% + filter(gender=="feminine") %% + mutate(height.feet_height/30.48) %% + filter/gender=='femin.gent_height.feet_height.feet_height.feet + filter/gender=='femin.gent_height.feet_height/30.48) %% + mutate(height.feet_height/30.48) %% + mutate(height.feet_height/30.48) %% + mutate(height.feet_height/30.48) %% + filter/gender=='femin.gent_height.feet	[1] "name" "height" "mass" "hair_color" "skin_color" "eye_color" "birth_year" "sex" [9] dandar" "bomeworld" "renciae" "films" "vehiclas" "starshipe"	arrange (dplyr) R Documentation	
<pre>+ filt&amp;(gender=="feminie") %&gt;# + mutat(height.feet&gt;height)0.48) %&gt;# + mutat(height.feet&gt;height)0.48) %&gt;# + mutat(height.feet&gt;height)0.48) %&gt;# + mutat(height.feet&gt;height.acm.m=TRUE) * A tibbl 11 × 12 name chr&gt; chr&gt; chr&gt; chr&gt; chr&gt; chr&gt; chr&gt; chr&gt;</pre>	> starwars.cut %>%		
<pre>+ mutat (height. feet=height/30.48) %&gt;% + mutat (height. feet=height/30.48) %&gt;% + mutat (height. feet height. gen_mean_mean(height. feet har. m=TRUE)) %&gt;% R console – work interactively + filter height. mean_na.rm=TRUE) # A tibbla li × 12 name</pre>	+ filter(gender=="feminine") %>%	Order rows using column values	
<pre>mutate (ref)(::mear(na:mear(ne)) (ref)(:mear(na:mear(ne))) * A tibile ll x l2 name height mass hair_color skin_color eye_color birth_year sex gender homeworld height.feet height.mean c/m² </pre> c/m²  db =	+ mutat (height.feet=height/30.48) %% mutat (height man man (height fact no mutut)) % <b>B console</b> – work interactively	Order Tows using column values	
<pre># A tibble 11 x 12 name chrs chrs chrs chrs chrs chrs chrs chrs</pre>	+ filler Neight. feet > height.mean.na.rm=TRUE)	Description	
<pre>name height mass hair_color skin_color eye_color birth_year sex gender homeworld height. feet height. mass hair_color skin_color eye_color birth_year sex gender homeworld height. feet height. mass hair_color skin_color eye_color birth_year sex gender homeworld height. feet height. mass hair_color skin_color eye_color birth_year sex gender homeworld height. feet height. mass hair_color skin_color eye_color birth_year sex gender homeworld height. feet height. mass hair_color skin_color eye_color birth_year sex gender homeworld height. feet height. mass hair_color skin_color eye_color birth_year sex gender homeworld height. feet height. mass hair_color eye data frame by the values of selected columns. Unlike other dplyr verbs, arrange() largely ignores grouping; you need to explicitly mention group by them, and functions of variables are evaluated once per data frame, not once per group. TRUE) in order to group by them, and functions of variables are evaluated once per data frame, not once per group.</pre>	# A tibble 11 × 12	Description	
Addisectings       Addisecting       Addisecting       A	name height mass hair_color skin_color eye_color birth_year sex gender homeworld height.fet height.mean	arrange () orders the rows of a data frame by the values of selected columns.	Plots help
2 Ayla Secura       178 55       none       blue       hazel       48 fema femin Ryloth       5.11       5.12       5.11       5.13       5.11       5.40       5.41       5.40       5.41       5.40       5.41       5.40       5.41       5.40       5.41       5.40       5.41       5.40       5.41       5.40       5.	< cnr > < cn	Unlike other delyr verbs, a maange (), largely igneres grouping; you need to	1 tots, notp,
3 Adi Gallia       184 50 none       dark       blue       NA fema. femin Coruscant       6.04       5.40         4 Luminara U       50 56.2 black       yellow       blue       40 fema femin Mirial       5.58       5.40         6 Dormé       165 NA       brown       light       brown       NA fema femin Maboo       5.41       5.40         7 Zam Wesell       168 55       blonde       fair, gre yellow       NA fema femin Coruscant       5.48       5.40         9 Jocasta Nu       167 NA       white       fair       black       NA fema femin Shili       5.84       5.40         10 Shaak Ti       178 57 none       red, blue black       NA fema femin Naboo       5.41       5.40         10 Shaak Ti       165 45       brown       NA fema femin Shili       5.84       5.40         10 Shaak Ti       165 45       brown       light       brown       NA fema femin Nabot       5.41       5.40         10 Shaak Ti       165 45       brown       light       brown       S.41       5.40       s.41       s.40         11 Padmé Amid       165 45       brown       light       brown       S.41       5.40       s.40         11	2 Ayla Secura 178 55 none blue hazel 48 fema femin Ryloth 5.84 5.40	explicitly mention grouping variables (or use .by group = TRUE) in order to	naalvadaa
4 Luminara U       4 So 56.2 black       yellow       blue       58 fema femin Mirial       5.38       5.40         5 Barriss Of       106 50       black       yellow       blue       40 fema femin Mirial       5.45       5.40         6 Dormé       165 NA       brown       light       brown       NA fema femin Naboo       5.41       5.40         7 Zam Wesell       168 55       blonde       fair, gre yellow       NA fema femin Zolan       5.51       5.40         9 Jocasta Nu       167 NA       write       fair       blue       NA fema femin Coruscant       5.48       5.40         10 Shaak Ti       178 57       none       red, blue       black       NA fema femin Naboo       5.41       5.40         11 Padmé Amid       165 45       brown       light       brown       46 fema femin Naboo       5.41       5.40         >       Image (.data,, .by_group = FALSE)       ## \$3 method for class 'data.frame'         arrange(.data,, .by_group = FALSE, .locale = NULL)       Image (.data,, .by_group = FALSE, .locale = NULL)	3 Adi Gallia 184 50 none dark blue NA fema femin Coruscant 6.04 5.40	group by them, and functions of variables are evaluated once per data frame, not	packages
6 Dormé       165 NA       brown       light       brown       NA fema femin Mirar       5.47       5.40         7 Zam Wesell       168 S5       blonde       fair, gre yellow       NA fema femin Xaboo       5.51       5.40         8 Taun We       213 NA       none       grey       black       NA fema femin Kamino       6.99       5.40         9 Jocasta Nu       167 NA       write       fair       block       NA fema femin Coruscant       5.48       5.40         10 Shaak Ti       178 57       none       red, blue black       NA fema femin Naboo       5.41       5.40         11 Padmé Amid       165 45       brown       light       brown       46 fema femin Naboo       5.41       5.40	4 Luminara U., VO 56.2 black yellow blue 58 fema, femin, Mirial 5.58 5.40	once per group.	
7 Zam Wesell       168       5       blonde       fair, gre yellow       NA fema femin Zolan       5.51       5.40         8 Taun We       213       NA       none       grey       black       NA fema femin Kamino       6.99       5.40         9 Jocasta Nu       167       NA       white       fair       blue       NA fema femin Kamino       6.99       5.40         10 Shaak Ti       178       57       none       red, blue black       NA fema femin Shili       5.84       5.40         11 Padmé Amid       165       45       brown       Hé fema femin Nabor       5.41       5.40	6 Dormé 165 NA brown light brown NA fema. femin. Naboo 541 5.40	Usage	
8 Taun We       213 NA none       grey       black       NA fema femin Kamino       6.99       5.40         9 Jocasta Nu       167 NA       White       fair       blue       NA fema femin Coruscant       5.48       5.40         10 Shaak Ti       178 57 none       red, blue black       NA fema femin Shili       5.84       5.40         11 Padmé Amid       165 45       brown       light       brown       46 fema femin Nabor       5.41       5.40         >       -       -       -       -       -       -       -	7 Zam Wesell 168 St blonde fair, gre… yellow NA fema… femin… Zolan 5.51 5.40	ougo	
10 Shaak Ti       178 57       57       none       red, blue black       NA fema femin Shili       5.84       5.40         11 Padmé Amid       165 45       brown       light       brown       46 fema femin Nabor       5.41       5.40	8 Taun We 213 NA none grey black NA fema. femin. Kamino 6.99 5.40	arrange(.data,, .by_group = FALSE)	
11 Padmé Amid       165 45 brown       1ight       brown       46 fema femin Nabes       5.41       5.40       ## S3 method for class 'data.frame'         >         ## s3 method for class 'data.frame'         arrange(.data,, .by group = FALSE, .locale = NULL)	10 Shak Ti 178 57 none red, blue, black NA fema, femin, Shili 5,84 5,40		
<pre>&gt; arrange(.data,, .by_group = FALSE, .locale = NULL)</pre>	11 Padmé Amid 165 45 brown light brown 46 fema femin Nabor 5.41 5.40	## S3 method for class 'data.frame'	
	>	arrange(.data,, .by_group = FALSE, .locale = NULL)	

#### nvironment ariables

#### R Studio tour – live demonstration (tour 1)

RStudio		- 0 X
ile <u>E</u> dit <u>C</u> ode <u>V</u> iew <u>P</u> lots <u>S</u> ession <u>B</u> uild <u>D</u> ebug <u>P</u> rofile <u>T</u> ools <u>H</u> elp		
🕽 🔹 🧐 🚰 🔹 🔚 🔚 📄 📝 Go to file/function 🔤 🛛 🔀 🔹 Addins 👻		Project: (None) •
<pre>Prode for session 1 markdownRmd* P R session 1 tour 1.R X Source on Save P P P R session 1 tour 1.R X Source on Save P P P R session 1 tour 1.R X This is a comment The first we are going to see how to run code in R The first we are going to see how to run code in R The first we are going to see how to run code in R The first we are going to see how to run code in R The first we are going to see how to run code in R The first we are going to see how to run code in R The first we are going to see how to run code in R The first we are going to see how to run code in R The first we are going to see how to run code in R The first we are going to see how to run code in R The first we are going to create a couple of new variables The first</pre>	ered better form to use <-	Environment       History       Connections       Tutorial         Import Dataset *       850 MiB *       List *       *         R *       Global Environment *       *       *         Values       *       int [1:10] 1 2 3 4 5 6 7 8 9 10       *         y       int [1:10] 1 4 9 16 25 36 49 64       z       385L
<pre>is a console terminal x Render x Background Jobs x  R 4.3.3 · ~/R intensive/ </pre>	panel)  R Script :  R Script :	Files       Plots       Packages       Help       Viewer       Presentation         Image: Second Secon
<pre>&gt; ## &lt;- is the assignment operator. You can also use an equal sign (=) but considered &gt; print(x) [1] 1 2 3 4 5 6 7 8 9 10 &gt; # we can see that x is now a variable in the global environment (see upper right pand &gt; y &lt;- x*x &gt; # variable named y consisting of an integer vector is now also listed in the global of &gt; # we can also work interactively in the console [TRUNCATED] &gt; # see that the plot is now in the lower right panel &gt; plot(y) &gt; # we can scroll between the two plots we created &gt; </pre>	<pre> [TRUNCATED] el) environment</pre>	$ \begin{array}{c}                                     $

#### Some useful shortcuts

- To clear your console –> CNTL+L
- To clear your environment -> use the broom function
- To run a selected line from the source panel -> CNTL+ENTER
- To run the whole script from the source panel -> CNTL+ALT+R
- To repeat a command in the console use the <mark>^</mark> arrow

## Getting data into R

- In the first tour, I created vectors x and y within R, which is not the usual workflow
- Usually, you will *import an Excel spreadsheet* where you have collected and organized your data. We will show you how to do this.
- Today we will primarily work with data using a standard wide format unique observations in rows and variables in columns
- A *long* format is used when you have multiple observations for an individual over time or space

## Wide vs long data formats

#### Example of wide format:

ID	<b>x1</b>	x2	<b>x3</b>	x4
1	3.3	1	2	17
2	2.1	1	3	22
3	7.0	0	1	10
4	4.1	1	1	42

#### Example of long format:

ID	time	x1	x2	х3	x4
1	1	3.3	1	2	17
1	2	0.5	1	1	14
2	1	2.1	1	3	22
2	2	2.8	0	5	16
3	1	7.0	0	1	10
4	1	4.1	1	1	42
4	2	4.1	0	0	57
5	1	3.8	1	1	12

## Data frames/tibbles

- When you import your data set into R, you typically store your data in a **data frame** within R
  - A **dataFrame** in R is a tabular (i.e., 2-dimensional, rectangular) data structure used to store values of any data type
  - A **tibble** is a newer version of the data frame, used in tidyverse

## Installing and Loading R packages (libraries)

- Over 10,000 libraries to help you analyze data
- These libraries contain **R packages**, which are collections of R functions, data and compiled code
- R libraries have to be loaded every time you open an R session

## tidyverse

- **Base R** refers to all the functionality that comes built into the R programming language. It is what you get when you open up the R console for the first time.
- The <u>tidyverse</u> is a collection of packages that add onto R to allow you to manipulate your data intuitively. It emphasizes readability. <u>It does not</u> <u>replace base R.</u>
- *data.table* is another such collection of packages
- In this session, we focus on learning the basics of some main packages in tidyverse (dplyr and ggplot2)

## Installing and loading the tidyverse core

install.packages('tidyverse') #Note: you did this prior to the workshop
library('tidyverse')
library('dplyr') #Note: dplyr automatically gets installed with tidyverse

Within *tidyverse*, there is package called *dplyr*. Think of the **d** as standing for **data** and the **plyr** standing for **plyers** – the goal of this package is to manipulate data frames in useful ways.

We will be focusing on the *dplyr* package in R this morning. This afternoon, we will learn about another package called *ggplot2* 

## Live demonstration (tour 2)

#### RStudio

<u>File Edit Code View Plots Session Build Debug Profile Tools H</u>elp

0 - Q	🛐 🖙 🗣 🔚 🔚 📄 🍺 Go to file/function 👘 🗄 🔹 Addins 👻				🔋 Project: (None) 🝷
🔍 r co	de for session 1 markdown.Rmd* × 💽 R session 1 tour 1.R × 💿 R session tour 2.R ×		Environment Histor	ry Connections Tutorial	
$\langle \Box \Box \rangle$	🔊 🔚 🖸 Source on Save 🔍 🎢 🖌 📋	Run   笋 🗘 🕂 🖬 Source 🔹 🗏	💣 🔒 📅 Import D	Dataset 👻 🌗 293 MiB 👻 💰	🗏 List 🖌 🛛 📿 🗸
1	# Now let's load the packages we need		R 🔹 🛑 Global Enviro	onment 👻	Q,
3	library(tidyverse)		Data		
4			🔍 xy	10 obs. of 2 variables	
5	# scroll the packages in the lower right panel to see what is checked		Values		
7	# If we need help on a particular package that has already been loaded we can typ	e:	X	int [1:10] 1 2 3 4 5 6 7 8 9	10
8	D Ju ]		У 7	101 [1:10] 1 4 9 10 25 30 49	64 81 100
10	2 ap i yr		2	JOJE	
11					
12 13	# and look on the lower right panel (HELP) for more information about the dplyr p	ackage			
14 - 15					
16 17	# Create a data frame/tibble by binding x and y together				
18	<pre>xy &lt;- bind_cols(x,y)</pre>				
20	class(xy)				
22	<pre>colnames(xy)=c("x","y")</pre>		•		Þ
25	<pre>plot(xy\$x,xy\$y)</pre>		Files Plots Packa	ges Help Viewer Presentation	-0
25					Q (C)
			R: dplyr: A Grammar	r of Data Manipulation - Find in Topic	
			dplyr-package {dply	/r}	R Documentation
			dolvr: A Gra	ammar of Data Manipulati	on
1:1	(Top Level) 💲	R Script 💠	Description		
Consol	e Terminal × Render × Background Jobs ×		To learn more about	t dplyr, start with the vignettes: browseVigne	ttes(package =
R R	4.3.3 · ~/R intensive/ ↔	$\odot$ $\checkmark$	"dplyr")		
> ?up >	ıyı <sup>.</sup>		Author(s)		
			Maintainer: Hadley	Wickham hadley@posit.co (ORCID)	
			Authors:		
			Romain Fran	çois ( <u>ORCID</u> )	
			Lionel Henry		
			Kirill Müller (	ORCID)	•

## The main verbs of **dplyr**

I just used a simple dplyr function called **bind\_cols()** to combine vectors x and y, but this is only one of many. Below are the **main verbs** we will focus on this morning

- Operation on columns (variables)
  - select
  - mutate
  - rename
  - relocate
  - Pull
- Operations on rows (observations)
  - arrange
  - filter
  - slice\_min, slice\_max

- Grouping and summarizing (observations)
  - group\_by
  - summarize
- Joining two data frames
  - inner\_join, left\_join, right\_join, full\_join

#### Data transformation with dplyr :: **CHEATSHEET**

dplyr functions work with pipes and expect tidy data. In tidy data:



#### Summarize Cases

Apply summary functions to columns to create a new table of summary statistics. Summary functions take vectors as input and return one value (see back).

> summary function summarize(.data, ...)

Compute table of summaries. mtcars |> summarize(avg = mean(mpg))

> count(.data, ..., wt = NULL, sort = FALSE, name = NULL) Count number of rows in each group defined by the variables in ... Also tally(), add\_count(), add tally(). mtcars < count(cyl)

#### Group Cases

Use group\_by(.data, ..., .add = FALSE, .drop = TRUE) to create a "grouped" copy of a table grouped by columns in ... dplyr functions will manipulate each "group" separately and combine the results.



Use **rowwise(**.data, ...) to group data into individual rows. dplyr functions will compute results for each row. Also apply functions to list-columns. See tidyr cheat sheet for list-column workflow.

#### Manipulate Cases

#### EXTRACT CASES

→

Row functions return a subset of rows as a new table.

→ **•** filter(.data, ..., .preserve = FALSE) Extract rows that meet logical criteria. mtcars > filter(mpg > 20)

distinct(.data, ..., .keep\_all = FALSE) Remove rows with duplicate values. mtcars |> distinct(gear)

> slice(.data, ..., .preserve = FALSE) Select rows by position. mtcars |> slice(10:15)

slice\_sample(.data, ..., n, prop, weight\_by = NULL, replace = FALSE) Randomly select rows. Use n to select a number of rows and prop to select a fraction of rows.

mtcars |> slice\_sample(n = 5, replace = TRUE)

slice\_min(.data, order\_by, ..., n, prop, with ties = TRUE) and slice max() Select rows with the lowest and highest values. mtcars |> slice min(mpg, prop = 0.25)

slice\_head(.data, ..., n, prop) and slice\_tail() Select the first or last rows.  $mtcars \ge slice_head(n = 5)$ 

#### Logical and boolean operators to use with filter()

==	<	<=	is.na()	%in%		xor(
!=	>	>=	!is.na()	1	&	

See ?base::Logic and ?Comparison for help.

#### ARRANGE CASES



arrange(.data, ..., .by\_group = FALSE) Order rows by values of a column or columns (low to high was with dependent or from high to be

#### Manipulate Variables

#### EXTRACT VARIABLES

Column functions return a set of columns as a new vector or table.



pull(.data, var = -1, name = NULL, ...) Extract column values as a vector, by name or index. mtcars > pull(wt)



select(.data, ...) Extract columns as a table. mtcars > select(mpg, wt)



relocate(.data, ..., .before = NULL, .after = NULL) Move columns to new position. mtcars |> relocate(mpg, cyl, .after = last\_col())

#### Use these helpers with select() and across() e.g. mtcars |> select(mpg:cyl)

num\_range(prefix, range) contains(match) :, e.g., mpg:cyl all\_of(x)/any\_of(x, ..., vars) !, e.g., !gear matches(match) everything() ends with(match) starts with(match) matches(match)

#### MANIPULATE MULTIPLE VARIABLES AT ONCE

df <- tibble(x\_1 = c(1, 2), x\_2 = c(3, 4), y = c(4, 5))



**across(**.cols, .funs, ..., .names = NULL) Summarize or mutate multiple columns in the same way. df |> summarize(across(everything(), mean))



c\_across(.cols) Compute across columns in row-wise data. df |>

rowwise() > mutate(x total = sum(c across(1:2)))

#### MAKE NEW VARIABLES

Apply vectorized functions to columns. Vectorized functions take vectors as input and return vectors of the same length as output (see back).

## Example data set

We will use the starwars data set available in the dplyr package

This data set consists of 87 observations and 14 variables.

0 - 0 📾 - 🗆 🗂 📥					R Pr	niect.
P corrige 1 tour 1 P X     O P c	errien taur 3 D# X D data cete X D data cete X O D secriton taur 3 D# X O reads for secriton 1 markdown Dmd# X		Environment History	Connections Tutorial	<b>Sec.</b> 11	Jecu
SW1SS	SWISS FERTILITY AND SOCIOECONOMIC INDICATORS (1888) Data		Timport Datase	et 🔹 🔰 321 MiB 👻		list •
treering	Yearly Treering Data, -6000-1979	•	R 🔹 📑 Global Environme	nt 👻	Q,	
trees	Diameter, Height and Volume for Black Cherry Trees					
uspop	Populations Recorded by the US Census					
volcano	Topographic Information on Auckland's Maunga Whau Volcano			Environment is empty		
warpbreaks	The Number of Breaks in Yarn during Weaving					
women	Average Heights and Weights for American Women					
Data sets in package 'd	plyr':					
band_instruments	Band membership					
band_instruments2	Band membership					
band_members	Band membership					
starWdf5	Starwars Characters Storm tracks data					
2 CO 1 M 3	Storm Sriend data					
Data sets in package 'f	orcats':					
gss_cat	A sample of categorical variables from the General Social survey					
Data sets in package 'g	gplot2':					
diamonds	Prices of over 50,000 round cut diamonds					
economics	US economic time series					
economics_long	US economic time series		Files Plots Packages	Help Viewer Presentation		
faithfuld	2d density estimate of Old Faithful data			•	0	
luv_colours	'colors()' in Luv space	-	💟 Install 🛛 🖤 Update		4	
midwest	Midwest demographics		Name	Description	Version	
Console Terminal × Render	K Background Jobs X		sys	Powerful and Reliable Tools for Running System Commands in R	3.4.2	
R 4.3.3 · ~/R intensive/ 🖗		∞ <u>∕</u>		Contains Matthe Fact Finding		æ
data()			systemfonts	System induve Font Finding	1.0.5	
> data()			table1	Tables of Descriptive Statistics in HTML	1.0.5	
> data() >			systemfonts       table1       testthat	Tables of Descriptive Statistics in HTML Unit Testing for R	1.0.5 1.4.3 3.2.0	9
> data() >			systemfonts       table1       testthat       textshaping	Tables of Descriptive Statistics in HTML Unit Testing for R Bindings to the 'HarfBuzz' and 'Fribidi' Libraries for Text Shaping	1.0.5 1.4.3 3.2.0 0.3.7	
> data() >			systemionts       table1       testthat       textshaping       TH.data	Tables of Descriptive Statistics in HTML Unit Testing for R Bindings to the 'HarfBuzz' and 'Fribidi' Libraries for Text Shaping TH's Data Archive	1.0.5       1.4.3       3.2.0       0.3.7       1.1-2	
> data() >			systemionts       table1       testthat       textshaping       TH.data       ✓	Tables of Descriptive Statistics in HTML Unit Testing for R Bindings to the 'HarfBuzz' and 'Fribidi' Libraries for Text Shaping TH's Data Archive Simple Data Frames	1.0.5         1.4.3         3.2.0         0.3.7         1.1-2         3.2.1	
> data() >			systemionts       table1       testthat       textshaping       TH.data       ✓ tibble       ✓ tidyr	Tables of Descriptive Statistics in HTML Unit Testing for R Bindings to the 'HarfBuzz' and 'Fribidi' Libraries for Text Shaping TH's Data Archive Simple Data Frames Tidy Messy Data	1.0.5         1.4.3         3.2.0         0.3.7         1.1-2         3.2.1         1.3.0	
> data() >			systemionts       table1       testthat       TH.data       ✓ tibble       ✓ tidyr       tidyselect	System Native Fold Finding Tables of Descriptive Statistics in HTML Unit Testing for R Bindings to the 'HarfBuzz' and 'Fribidi' Libraries for Text Shaping TH's Data Archive Simple Data Frames Tidy Messy Data Select from a Set of Strings	1.0.5         1.4.3         3.2.0         0.3.7         1.1-2         3.2.1         1.3.0         1.2.0	
> data() >			systemionts       table1       testthat       textshaping       TH.data       ✓ tibble       ✓ tidyr       tidyselect       ✓ tidyverse	System Native Fort Finding Tables of Descriptive Statistics in HTML Unit Testing for R Bindings to the 'HarfBuzz' and 'Fribidi' Libraries for Text Shaping TH's Data Archive Simple Data Frames Tidy Messy Data Select from a Set of Strings Easily Install and Load the 'Tidyverse'	1.0.5 1.4.3 3.2.0 0.3.7 1.1-2 3.2.1 1.3.0 1.2.0 2.0.0	
> data() >			systemionts       table1       testthat       textshaping       TH.data       ✓ tibble       ✓ tidyr       tidyselect       ✓ tidyverse       timechange	System Native Fold Finding Tables of Descriptive Statistics in HTML Unit Testing for R Bindings to the 'HarfBuzz' and 'Fribidi' Libraries for Text Shaping TH's Data Archive Simple Data Frames Tidy Messy Data Select from a Set of Strings Easily Install and Load the 'Tidyverse' Efficient Manipulation of Date-Times	1.0.5 1.4.3 3.2.0 0.3.7 1.1-2 3.2.1 1.3.0 1.2.0 2.0.0 0.2.0	
> data() >			systemtonts         table1         testthat         textshaping         TH.data         tibble         tidyr         tidyselect         tidyverse         timechange         timeDate	System Nature Fohr Finding Tables of Descriptive Statistics in HTML Unit Testing for R Bindings to the 'HarfBuzz' and 'Fribidi' Libraries for Text Shaping TH's Data Archive Simple Data Frames Tidy Messy Data Select from a Set of Strings Easily Install and Load the 'Tidyverse' Efficient Manipulation of Date-Times Rmetrics - Chronological and Calendar Objects	1.0.5 1.4.3 3.2.0 0.3.7 1.1-2 3.2.1 1.3.0 1.2.0 2.0.0 0.2.0 4032.109	
> data() >			systemionts         table1         testthat         textshaping         TH.data         tibble         tidyr         tidyselect         tidyverse         timechange         timeDate         tinytex	System Native For Finding Tables of Descriptive Statistics in HTML Unit Testing for R Bindings to the 'HarfBuzz' and 'Fribidi' Libraries for Text Shaping TH's Data Archive Simple Data Frames Tidy Messy Data Select from a Set of Strings Easily Install and Load the 'Tidyverse' Efficient Manipulation of Date-Times Rmetrics - Chromological and Calendar Objects Helper Functions to Install and Maintain TeX Live, and Compile LaTeX Documents	1.0.5 1.4.3 3.2.0 0.3.7 1.1-2 3.2.1 1.3.0 1.2.0 2.0.0 0.2.0 4032.109 0.48	
> data() >			systemionts         table1         testthat         textshaping         TH.data         tibble         tidyr         tidyselect         timechange         timepate         tinytex	System Native Fold Finding Tables of Descriptive Statistics in HTML Unit Testing for R Bindings to the 'HarfBuzz' and 'Fribidi' Libraries for Text Shaping TH's Data Archive Simple Data Frames Tidy Messy Data Select from a Set of Strings Easily Install and Load the 'Tidyverse' Efficient Manipulation of Date-Times Rmetrics - Chronological and Calendar Objects Helper Functions to Install and Maintain TeX Live, and Compile LaTeX Documents Time Zone Database Information	1.05 1.4.3 3.2.0 0.3.7 1.1-2 3.2.1 1.3.0 1.2.0 2.0.0 0.2.0 4032.109 0.48	

#### Example data set

#### > glimpse(starwars)

Rows: 87

#### Columns: 14

<chr> "Luke Skywalker", "C-3PO", "R2-D2", "Darth Vader", "Leia Organa", "Owen Lars", "Beru Whitesun lars", "R5-D4", "Biggs Da... \$ name <int> 172, 167, 96, 202, 150, 178, 165, 97, 183, 182, 188, 180, 228, 180, 173, 175, 170, 180, 66, 170, 183, 200, 190, 177, 17... \$ height <db1> 77.0, 75.0, 32.0, 136.0, 49.0, 120.0, 75.0, 32.0, 84.0, 77.0, 84.0, NA, 112.0, 80.0, 74.0, 1358.0, 77.0, 110.0, 17.0, 7... \$ mass \$ hair\_color <chr> "blond", NA, NA, "none", "brown", "brown, grey", "brown", NA, "black", "auburn, white", "blond", "auburn, grey", "brown... \$ skin\_color <chr> "fair", "gold", "white, blue", "white", "light", "light", "light", "white, red", "light", "fair", "fair", "fair", "unkn... \$ eye\_color <chr> "blue", "yellow", "red", "yellow", "brown", "blue", "blue", "red", "brown", "blue-gray", "blue", "blue", "blue", "brown... \$ birth\_year <db7> 19.0, 112.0, 33.0, 41.9, 19.0, 52.0, 47.0, NA, 24.0, 57.0, 41.9, 64.0, 200.0, 29.0, 44.0, 600.0, 21.0, NA, 896.0, 82.0,... <chr> "male", "none", "none", "male", "female", "female", "female", "none", "male", "male \$ sex <chr> "masculine", "masculine", "masculine", "masculine", "feminine", "masculine", "feminine", "masculine", "masculine, "masculine", "masculine, \$ gender <chr> "Tatooine", "Tatooine", "Naboo", "Tatooine", "Alderaan", "Tatooine", "Tatooine", "Tatooine", "Tatooine", "Stewjon", "Ta... \$ homeworld <chr> "Human", "Droid", "Droid", "Human", "Human", "Human", "Human", "Droid", "Human", "Human", "Human", "Human", "Wookiee", ... \$ species //ist> <"The Empire Strikes Back", "Revenge of the Sith", "Return of the Jedi", "A New Hope", "The Force Awakens">, <"The Emp...</pre> \$ films \$ vehicles  $<1ist> <"X-wing", "Imperial shuttle">, <math>\diamond$ ,  $\diamond$ , "TIE Advanced x1",  $\diamond$ ,  $\diamond$ ,  $\diamond$ ,  $\diamond$ ,  $\diamond$ , "X-wing", <"Jedi starfighter", "Trade Feder... \$ starships

#### 87 rows x 14 columns

## **Column operations**

## dplyr::select

• Select (and optionally rename) variables in a data frame

Example:

>starwars.cut <- select(starwars,c(1:6,8:10)) #dplyr</pre>

> starwars.cut <- starwars[,c(1:6,8:10)] #base R</pre>

### dplyr::select

Note: these are the same, though assignment operator (<-) preferred

>starwars.cut <- select(starwars,c(1:6,8:10))
>starwars.cut = select(starwars,c(1:6,8:10))

*Note: these are NOT the same* 

>starwars.cut <- select(starwars,c(1:6,8:10))
>starwars\_cut = select(starwars,c(1:6,8:10))

# Can't exactly remember how the function works, key words, or options?

#### > ?select

soloct (dplyr)	P. Decumentat
Select {upiyi}	R Documental
Keep or drop columns using	g their names and type
Description	
Select (and optionally rename) variables in a data makes it easy to refer to variables based on their on the left to f on the right) or type (e.g. where ( columns).	a frame, using a concise mini-language t name (e.g. a:f selects all columns from is.numeric) selects all numeric
Overview of selection features	
Tidyverse selections implement a dialect of R who variables:	ere operators make it easy to select
for selecting a range of consecutive varia	ables.
I for taking the complement of a set of var	iables.
+ ${\scriptstyle\&}$ and ${\scriptstyle }$ for selecting the intersection or the	e union of two sets of variables.
• c() for combining selections.	
In addition, you can use <b>selection helpers</b> . Some	e helpers select specific columns:
<ul> <li><u>everything()</u>: Matches all variables.</li> </ul>	
<ul> <li><u>last_col()</u>: Select last variable, possibly</li> </ul>	y with an offset.
<ul> <li><u>group_cols()</u>: Select all grouping column</li> </ul>	ins.
Other helpers select variables by matching patter	ns in their names:
<ul> <li><u>starts_with()</u>: Starts with a prefix.</li> </ul>	
<ul> <li><u>ends_with()</u>: Ends with a suffix.</li> </ul>	
<ul> <li><u>contains()</u>: Contains a literal string.</li> </ul>	

#### Practice and learn:

#### Create a small data set and play around with different options

```
Console Terminal ×
                  Render ×
                          Background Jobs ×
R 4.3.3 · ~/R intensive/ 🖉
> example.data<-data.frame(cbind(c('A', 'B', 'C'), c(1,2,3), c(0, NA, NA)))</pre>
> example.data
  X1 X2 X3
  A 1
           0
  B 2 <NA>
2
3 C 3 <NA>
> select(example.data,1)
  X1
1 A
2 B
3 C
> select(example.data,1:2)
 X1 X2
1 A 1
2 B 2
3 C 3
> select(example.data,contains("3"))
    X3
   0
1
2 <NA>
3 <NA>
>
```

#### Example data set

>head(starwars.cut) #base R function to print first few rows of a data frame

##	#	A tibble: (	6 x 9							
##		name	height	mass	hair_color	skin_color	eye_color	sex	gender	homeworld
##		<chr></chr>	<int></int>	<dbl></dbl>	<chr></chr>	<chr></chr>	<chr></chr>	<chr></chr>	<chr></chr>	<chr></chr>
##	1	Luke Skyw~	172	77	blond	fair	blue	male	mascu~	Tatooine
##	2	C-3PO	167	75	<na></na>	gold	yellow	none	mascu~	Tatooine
##	3	R2-D2	96	32	<na></na>	white, bl~	red	none	mascu~	Naboo
##	4	Darth Vad~	202	136	none	white	yellow	male	mascu~	Tatooine
##	5	Leia Orga~	150	49	brown	light	brown	fema~	femin~	Alderaan
##	6	Owen Lars	178	120	brown, gr~	light	blue	male	mascu~	Tatooine

A tibble 6x 9 – a tibble is a tidyverse data frame. All dplyr verbs take a tibble as input

#### dplyr::mutate

#### • Used to replace/update the values of columns

#### Suppose we want to compute height in feet instead of meters >mutate(starwars.cut, height.feet = height/30.48)

	name	height	mass	hair_co	lor	skin_co	lor	eye_color	birth_ye	ar se	ex	gender	homeworld	height.feet
	<chr></chr>	<int></int>	<db1></db1>	<chr></chr>		<chr></chr>		<chr></chr>	<db< td=""><td>7&gt; &lt;0</td><td>chr&gt;</td><td><chr></chr></td><td><chr></chr></td><td><db1></db1></td></db<>	7> <0	chr>	<chr></chr>	<chr></chr>	<db1></db1>
1	Luke Skywalker	172	77	blond		fair		blue	19	ma	ale	mascu	Tatooine	5.64
2	C-3PO	167	75	NA		gold		yellow	112	no	one	mascu	Tatooine	5.48
3	R2-D2	96	32	NA		white,	blue	red	33	no	one	mascu	Naboo	3.15
4	Darth Vader	202	136	none		white		yellow	41	.9 ma	ale	mascu	Tatooine	6.63
5	Leia Organa	150	49	brown		light		brown	19	fe	emale	femin	Alderaan	4.92
6	Owen Lars	178	120	brown,	grey	light		blue	52	ma	ale	mascu	Tatooine	5.84
7	Beru Whitesun lars	165	75	brown		light		blue	47	fe	emale	femin	Tatooine	5.41
8	R5-D4	97	32	NA		white,	red	red	NA	no	one	mascu	Tatooine	3.18
9	Biggs Darklighter	183	84	black		light		brown	24	ma	ale	mascu	Tatooine	6.00
10	Obi-Wan Kenobi	182	77	auburn,	white	fair		blue-gray	57	ma	ale	mascu	Stewjon	5.97
# i	77 more rows													

## Combining verbs with pipes (%>%)

>mutate(starwars.cut, height.feet = height/30.48)

is the same as:

>starwars.cut %>% mutate(height.feet = height/30.48)

"take the starwars.cut data set **then** add a new variable with height in feet"

Why use pipes? It can make code easy to follow and can avoid repetitive typing of (for example) the data frame name in every function

Shortcut to type %>%: CNTL+SHIFT+M

## Combining verbs with pipes (%>%)

If we instead want to create a new data frame with height.feet included:

> starwars.cut1 = starwars %>% mutate(height.feet = height/30.48)

We can now look on the top right (environment) to see this new data frame

#### dplyr::rename, rename\_with

• Changes the name of a column

>rename(starwars.cut1,eye.color=eye\_color)

>starwars.upper <- rename\_with(starwars.cut1,toupper)</pre>

#### dplyr::relocate

• Changes column positions

#### > relocate(starwars.upper,MASS,.after=last\_col())

# A	tibble: 87 × 11													
N	AME	HEIGHT	HAIR_CO	OLOR	SKIN_CO	DLOR	EYE_COLOR	BIRTH_YE	EAR	SEX	GENDER	HOMEWORLD	HEIGHT.FEET	MASS
<	chr>	<int></int>	<chr></chr>		<chr></chr>		<chr></chr>	<dl< td=""><td>b7&gt;</td><td><chr></chr></td><td><chr></chr></td><td><chr></chr></td><td><db1></db1></td><td><db1></db1></td></dl<>	b7>	<chr></chr>	<chr></chr>	<chr></chr>	<db1></db1>	<db1></db1>
1 L	uke Skywalker	172	blond		fair		blue	19	9	male	masculine	Tatooine	5.64	77
2 C	-3P0	167	NA		gold		yellow	112	2	none	masculine	Tatooine	5.48	75
3 R	2-D2	96	NA		white,	blue	red	33	3	none	masculine	Naboo	3.15	32
4 D	arth Vader	202	none		white		yellow	41	L.9	male	masculine	Tatooine	6.63	136
5 L	eia Organa	150	brown		light		brown	19	9	female	feminine	Alderaan	4.92	49
6 O	wen Lars	178	brown,	grey	light		blue	52	2	male	masculine	Tatooine	5.84	120
7 B	eru Whitesun lars	165	brown		light		blue	47	7	female	feminine	Tatooine	5.41	75
8 R	5-D4	97	NA		white,	red	red	NA	4	none	masculine	Tatooine	3.18	32
9 B	iggs Darklighter	183	black		light		brown	24	1	male	masculine	Tatooine	6.00	84
10 <b>O</b>	bi-Wan Kenobi	182	auburn	, white	fair		blue-gray	57	7	male	masculine	Stewjon	5.97	77
<b># i</b> 7	7 more rows													

# i Use `print(n = ...)` to see more rows

## dplyr::pull

Extracts a variable (column) as a vector

#### >pull(starwars.cut,height)

>pull(starwars.cut,2)

#### > pull(starwars.cut,height)

[1] 172 167 96 202 150 178 165 97 183 182 188 180 228 180 173 175 170 180 66 170 183 200 190 177 175 180 150 NA 88 160 193 191 170 196 [35] 224 206 183 137 112 183 163 175 180 178 94 122 163 188 198 196 171 184 188 264 188 196 185 157 183 183 170 166 165 193 191 183 168 198 [69] 229 213 167 79 96 193 191 178 216 234 188 178 206 NA NA NA NA NA 165 > pull(starwars.cut,2)

[1] 172 167 96 202 150 178 165 97 183 182 188 180 228 180 173 175 170 180 66 170 183 200 190 177 175 180 150 NA 88 160 193 191 170 196 [35] 224 206 183 137 112 183 163 175 180 178 94 122 163 188 198 196 171 184 188 264 188 196 185 157 183 183 170 166 165 193 191 183 168 198 [69] 229 213 167 79 96 193 191 178 216 234 188 178 206 NA NA NA NA NA 165

## **Row operations**

## dplyr::arrange

- arrange orders the rows of a data frame by the values of selected columns
- Note: We can find more information using
- **??dplyr::arrange** (or **??arrange**) to get relevant help pages

### Example using arrange

- > starwars.cut %>% arrange(desc(height))
- > arrange(starwars.cut, desc(height))

#### The 87x10 tibble starwars.cut is to be sorted by height



> starwars.cut	%>% arra	unge (de	esc(height))	)					
# A tibble: 87	× 10								
name	height	mass	hair_color	skin_color	eye_color	birth_year	sex	gender	homeworld
<chr></chr>	<int></int>	<db1></db1>	<chr></chr>	<chr></chr>	<chr></chr>	<db1></db1>	<chr></chr>	<chr></chr>	<chr></chr>
$\rightarrow 1$ Yarael Poof	264	NA	none	white	yellow	NA	male	masculine	Quermia
2 Tarfful	234	136	brown	brown	blue	NA	male	masculine	Kashyyyk
3 Lama Su	229	88	none	grey	black	NA	male	masculine	Kamino
4 Chewbacca	228	112	brown	unknown	blue	200	male	masculine	Kashyyyk
5 Roos Tarpals	224	82	none	grey	orange	NA	male	masculine	Naboo
6 Grievous	216	159	none	brown, white	green, yellow	NA	male	masculine	Kalee
7 Taun We	213	NA	none	grey	black	NA	female	feminine	Kamino
8 Rugor Nass	206	NA	none	green	orange	NA	male	masculine	Naboo
9 Tion Medon	206	80	none	grey	black	NA	male	masculine	Utapau
10 Darth Vader	202	136	none	white	yellow	41.9	male	masculine	Tatooine
# i 77 more rows	5								

# i Use `print(n = ...)` to see more rows

### Example using arrange

• We can also sort by more than one variable

>starwars.cut %>% arrange(gender, desc(height))

which sorts by gender and then ascending height

> starwars.cut %>% group\_by(gender) %>%
 arrange(desc(height),.by\_group=TRUE)

accomplishes the same thing

## dplyr::filter

- used to subset a data frame, retaining all rows that satisfy your conditions
- Useful functions:
  - == > <
  - & | ! xor()
  - is.na()
  - between()
  - near()

#### Examples using *filter*

#### > starwars.cut %>% filter(homeworld=="Naboo")

	name	height	mass	hair_color	skin_color	eye_color	birth_year	sex	gender	homeworld
	<chr></chr>	<int></int>	<db1></db1>	<chr></chr>	<chr></chr>	<chr></chr>	<db7></db7>	<chr></chr>	<chr></chr>	<chr></chr>
1	R2-D2	96	32	NA	white, blue	red	33	none	masculine	Naboo
2	Palpatine	170	75	grey	pale	yellow	82	male	masculine	Naboo
3	Jar Jar Binks	196	66	none	orange	orange	52	male	masculine	Naboo
4	Roos Tarpals	224	82	none	grey	orange	NA	male	masculine	Naboo
5	Rugor Nass	206	NA	none	green	orange	NA	male	masculine	Naboo
6	Ric Olié	183	NA	brown	fair	blue	NA	NA	NA	Naboo
7	Quarsh Panaka	183	NA	black	dark	brown	62	NA	NA	Naboo
8	Gregar Typho	185	85	black	dark	brown	NA	male	masculine	Naboo
9	Cordé	157	NA	brown	light	brown	NA	female	feminine	Naboo
10	Dormé	165	NA	brown	light	brown	NA	female	feminine	Naboo
11	Padmé Amidala	165	45	brown	light	brown	46	female	feminine	Naboo

## Examples using *filter*

> starwars.cut %>%
filter(homeworld=="Naboo") %>%
filter(hair\_color != 'grey')

Note we can use a double ("Naboo") or single ('grey') quote interchangeably in R

	name	height	mass	hair_color	skin_color	eye_color	birth_year	sex	gender	homeworld
	<chr></chr>	<int></int>	<db1></db1>	<chr></chr>	<chr></chr>	<chr></chr>	<db1></db1>	<chr></chr>	<chr></chr>	<chr></chr>
1	Jar Jar Binks	196	66	none	orange	orange	52	male	masculine	Naboo
2	Roos Tarpals	224	82	none	grey	orange	NA	male	masculine	Naboo
3	Rugor Nass	206	NA	none	green	orange	NA	male	masculine	Naboo
4	Ric Olié	183	NA	brown	fair	blue	NA	NA	NA	Naboo
5	Quarsh Panaka	183	NA	black	dark	brown	62	NA	NA	Naboo
6	Gregar Typho	185	85	black	dark	brown	NA	male	masculine	Naboo
7	Cordé	157	NA	brown	light	brown	NA	female	feminine	Naboo
8	Dormé	165	NA	brown	light	brown	NA	female	feminine	Naboo
9	Padmé Amidala	165	45	brown	light	brown	46	female	feminine	Naboo

#### More succinctly, we can type:

>filter(starwars.cut,homeworld =="Naboo" & hair\_color != 'grey' )

### Examples using *filter*

#### > starwars.cut %>% filter(height > 200 ) %>% filter(mass != 'NA')

#### > starwars.cut %>% filter(height > 200 ) %>% filter(mass != 'NA')

# A tibble: 7 x 10

name	height	mass	hair_color	skin_color	eye_color	birth_year	sex	gender	homeworld
<chr></chr>	<int></int>	<db1></db1>	<chr></chr>	<chr></chr>	<chr></chr>	<db1></db1>	<chr></chr>	<chr></chr>	<chr></chr>
1 Darth Vader	202	136	none	white	yellow	41.9	male	masculine	Tatooine
2 Chewbacca	228	112	brown	unknown	blue	200	male	masculine	Kashyyyk
3 Roos Tarpals	224	82	none	grey	orange	NA	male	masculine	Naboo
4 Lama Su	229	88	none	grey	black	NA	male	masculine	Kamino
5 Grievous	216	159	none	brown, white	green, yellow	NA	male	masculine	Kalee
6 Tarfful	234	136	brown	brown	blue	NA	male	masculine	Kashyyyk
7 Tion Medon	206	80	none	grey	black	NA	male	masculine	Utapau

## slice\_min, slice\_max

#### > slice\_min(starwars.cut,order\_by=height,by=sex,na\_rm=TRUE) # A tibble: 7 × 10

	name	height	mass	hair_color	skin_color	eye_color	birth_year	sex	gender	homeworld
	<chr></chr>	<int></int>	<db1></db1>	<chr></chr>	<chr></chr>	<chr></chr>	<db1></db1>	<chr></chr>	<chr></chr>	<chr></chr>
1	Yoda	66	17	white	green	brown	896	male	masculine	NA
2	R2-D2	96	32	NA	white, blue	red	33	none	masculine	Naboo
3	R4-P17	96	NA	none	silver, red	red, blue	NA	none	feminine	NA
4	Leia Organa	150	49	brown	light	brown	19	female	feminine	Alderaan
5	Mon Mothma	150	NA	auburn	fair	blue	48	female	feminine	Chandrila
6	Jabba Desilijic Tiure	175	<u>1</u> 358	NA	green-tan, brow	n orange	600	hermaphroditic	masculine	Nal Hutta
7	Sly Moore	178	48	none	pale	white	NA	NA	NA	Umbara
>	slice_max(starwars.cut	.,order_	_by=he	ight,by=sex,	,na_rm=TRUE)					
#	A tibble: 6 × 10									
	name	height	mass	hair_color	skin_color	eye_color	birth_year	sex	gender	homeworld
	<chr></chr>	<int></int>	<db1></db1>	<chr></chr>	<chr></chr>	<chr></chr>	<db1></db1>	<chr></chr>	<chr></chr>	<chr></chr>
1	Yarael Poof	264	NA	none	white	yellow	NA	male	masculine	Quermia
2	IG-88	200	140	none	metal	red	15	none	masculine	NA
3	Taun We	213	NA	none	grey	black	NA	female	feminine	Kamino
4	Jabba Desilijic Tiure	175	<u>1</u> 358	NA	green-tan, brow	n orange	600	hermaphroditic	masculine	Nal Hutta
5	Ric Olié	183	NA	brown	fair	blue	NA	NA	NA	Naboo
6	Quarsh Panaka	183	NA	black	dark	brown	62	NA	NA	Naboo
	•									

## Grouping operations

## dplyr::group\_by

- Groups the data frame by a grouping variable(s). Becomes a grouped tibble where operations are performed by group until it is ungrouped
- NOT the same as sorting or *dplyr::arrange*

> group\_by(starwars.cut1, gender)
>ungroup(starwars.cut1)

## group\_by

#### > group\_by(starwars.cut1, gender)

- # A tibble: 87 × 11
- # Groups: gender [3]

name	height	mass	hair_color	skin_color	eye_color	birth_year	sex	gender	homeworld	height.feet
<chr></chr>	<int></int>	<db1></db1>	<chr></chr>	<chr></chr>	<chr></chr>	<db1></db1>	<chr></chr>	<chr></chr>	<chr></chr>	<db7></db7>
1 Luke Skywalker	172	77	blond	fair	blue	19	male	masculine	Tatooine	5.64
2 C-3PO	167	75	NA	gold	yellow	112	none	masculine	Tatooine	5.48
3 R2-D2	96	32	NA	white, blue	red	33	none	masculine	Naboo	3.15
4 Darth Vader	202	136	none	white	yellow	41.9	male	masculine	Tatooine	6.63
5 Leia Organa	150	49	brown	light	brown	19	female	feminine	Alderaan	4.92
6 Owen Lars	178	120	brown, grey	light	blue	52	male	masculine	Tatooine	5.84
7 Beru Whitesun lars	165	75	brown	light	blue	47	female	feminine	Tatooine	5.41
8 R5-D4	97	32	NA	white, red	red	NA	none	masculine	Tatooine	3.18
9 Biggs Darklighter	183	84	black	light	brown	24	male	masculine	Tatooine	6.00
10 Obi-Wan Kenobi	182	77	auburn, whit	e fair	blue-gray	57	male	masculine	Stewjon	5.97
# i 77 more rows										
11 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	N 1 4 4 4									
# i Use print(n =	) to s	ee mor	e rows							
<pre># i Use print(n = &gt; ungroup(starwars.cu</pre>	) to so t1)	ee mor	e rows							
<pre># 1 Use print(n = &gt; ungroup(starwars.cu # A tibble: 87 × 11</pre>	to so t1)	ee mor	e rows							
<pre># 1 Use print(n = &gt; ungroup(starwars.cu # A tibble: 87 × 11</pre>	t1) height	ee mor mass	e rows hair_color	skin_color	eye_color	birth_year	sex	gender	homeworld	height.feet
<pre># 1 Use print(n = &gt; ungroup(starwars.cu # A tibble: 87 × 11</pre>	to so t1) height <int></int>	ee mor mass <db1></db1>	e rows hair_color <i><chr></chr></i>	skin_color <chr></chr>	eye_color < <i>chr&gt;</i>	birth_year <db1></db1>	sex <chr></chr>	gender <chr></chr>	<pre>homeworld <chr></chr></pre>	height.feet <db1></db1>
<pre># 1 Use print(n = &gt; ungroup(starwars.cu # A tibble: 87 × 11</pre>	t1) height <i><int></int></i> 172	ee mor mass <i><db1></db1></i> 77	e rows hair_color <i><chr></chr></i> blond	skin_color <i><chr></chr></i> fair	eye_color <i><chr></chr></i> blue	birth_year <i><db< i="">7&gt; 19</db<></i>	sex <i><chr></chr></i> male	gender <i><chr></chr></i> masculine	homeworld <i><chr></chr></i> Tatooine	height.feet <i><db1></db1></i> 5.64
<pre># 1 Use print(n = &gt; ungroup(starwars.cu # A tibble: 87 × 11</pre>	t1) height <int> 172 167</int>	ee mor mass <db7> 77 75</db7>	e rows hair_color <i><chr></chr></i> blond NA	skin_color <i><chr></chr></i> fair gold	eye_color <i><chr></chr></i> blue yellow	birth_year <i><db1></db1></i> 19 112	sex <i><chr></chr></i> male none	gender < <i>chr&gt;</i> masculine masculine	homeworld <i><chr></chr></i> Tatooine Tatooine	height.feet <i><db1></db1></i> 5.64 5.48
<pre># 1 Use print(n = &gt; ungroup(starwars.cu # A tibble: 87 × 11</pre>	t1) height <i><int></int></i> 172 167 96	mass <db7> 77 75 32</db7>	e rows hair_color <i><chr></chr></i> blond NA NA	skin_color <i><chr></chr></i> fair gold white, blue	eye_color < <i>chr&gt;</i> blue yellow red	birth_year <i><db1></db1></i> 19 112 33	sex <i><chr></chr></i> male none none	gender < <i>chr&gt;</i> masculine masculine masculine	homeworld <i><chr></chr></i> Tatooine Tatooine Naboo	height.feet <i><db1></db1></i> 5.64 5.48 3.15
<pre># 1 Use print(n = &gt; ungroup(starwars.cu # A tibble: 87 × 11     name         <chr> 1 Luke Skywalker 2 C-3P0 3 R2-D2 4 Darth Vader</chr></pre>	t1) height <int> 172 167 96 202</int>	mass <db1> 77 75 32 136</db1>	e rows hair_color <i><chr></chr></i> blond NA NA none	skin_color <i><chr></chr></i> fair gold white, blue white	eye_color <chr> blue yellow red yellow</chr>	birth_year <i><db1></db1></i> 19 112 33 41.9	sex <i><chr></chr></i> male none none male	gender <chr> masculine masculine masculine masculine</chr>	homeworld <i><chr></chr></i> Tatooine Tatooine Naboo Tatooine	height.feet <db7> 5.64 5.48 3.15 6.63</db7>
<pre># 1 Use print(n = &gt; ungroup(starwars.cu # A tibble: 87 × 11     name         <chr> 1 Luke Skywalker 2 C-3P0 3 R2-D2 4 Darth Vader 5 Leia Organa</chr></pre>	t1) height <int> 172 167 96 202 150</int>	mass <db1> 77 75 32 136 49</db1>	e rows hair_color < <i>chr&gt;</i> blond NA NA none brown	skin_color <i><chr></chr></i> fair gold white, blue white light	eye_color <chr> blue yellow red yellow brown</chr>	birth_year <i><db1></db1></i> 19 112 33 41.9 19	sex <chr> male none none male female</chr>	gender <chr> masculine masculine masculine masculine feminine</chr>	homeworld <i><chr></chr></i> Tatooine Tatooine Naboo Tatooine Alderaan	height.feet <db1> 5.64 5.48 3.15 6.63 4.92</db1>
<pre># 1 Use print(n = &gt; ungroup(starwars.cu # A tibble: 87 × 11     name         <chr> 1 Luke Skywalker 2 C-3P0 3 R2-D2 4 Darth Vader 5 Leia Organa 6 Owen Lars</chr></pre>	t1) height <int> 172 167 96 202 150 178</int>	mass <db1> 77 75 32 136 49 120</db1>	e rows hair_color < <i>chr&gt;</i> blond NA NA none brown brown, grey	skin_color <i><chr></chr></i> fair gold white, blue white light light	eye_color < <u>chr&gt;</u> blue yellow red yellow brown blue	birth_year <i><db1></db1></i> 19 112 33 41.9 19 52	sex < <u>chr&gt;</u> male none none male female male	gender < <i>chr&gt;</i> masculine masculine masculine feminine masculine feminine	homeworld <i><chr></chr></i> Tatooine Tatooine Naboo Tatooine Alderaan Tatooine	height.feet < <u>db</u> 1> 5.64 5.48 3.15 6.63 4.92 5.84
<pre># 1 Use print(n = &gt; ungroup(starwars.cur # A tibble: 87 × 11     name         <chr> 1 Luke Skywalker 2 C-3P0 3 R2-D2 4 Darth Vader 5 Leia Organa 6 Owen Lars 7 Beru Whitesun lars</chr></pre>	height <int> 172 167 96 202 150 178 165</int>	mass <db7> 77 75 32 136 49 120 75</db7>	e rows hair_color < <i>chr&gt;</i> blond NA NA none brown brown, grey brown	skin_color <i><chr></chr></i> fair gold white, blue white light light light	eye_color <chr> blue yellow red yellow brown blue blue</chr>	birth_year <i><db1></db1></i> 19 112 33 41.9 19 52 47	sex <chr> male none male female female female</chr>	gender <chr> masculine masculine masculine feminine masculine feminine feminine</chr>	homeworld <i><chr></chr></i> Tatooine Tatooine Naboo Tatooine Alderaan Tatooine Tatooine	height.feet <dbl> 5.64 5.48 3.15 6.63 4.92 5.84 5.41</dbl>
<pre># 1 Use print(n = &gt; ungroup(starwars.cur # A tibble: 87 × 11     name         <chr> 1 Luke Skywalker 2 C-3P0 3 R2-D2 4 Darth Vader 5 Leia Organa 6 Owen Lars 7 Beru Whitesun lars 8 R5-D4</chr></pre>	height <int> 172 167 96 202 150 178 165 97</int>	mass <db7> 77 75 32 136 49 120 75 32</db7>	e rows hair_color <chr> blond NA NA none brown brown, grey brown NA</chr>	skin_color <chr> fair gold white, blue white light light light white, red</chr>	eye_color <chr> blue yellow red yellow brown blue blue red</chr>	birth_year <db7> 19 112 33 41.9 19 52 47 NA</db7>	sex <chr> male none male female female female none</chr>	gender < <i>chr&gt;</i> masculine masculine masculine feminine masculine feminine masculine feminine masculine	homeworld <chr> Tatooine Tatooine Naboo Tatooine Alderaan Tatooine Tatooine Tatooine Tatooine</chr>	height.feet <dbl> 5.64 5.48 3.15 6.63 4.92 5.84 5.41 3.18</dbl>

blue-gray

57

77 auburn, white fair

male masculine Stewjon

5.97

10 Obi-Wan Kenobi

# i 77 more rows

# i Use `print(n = ...)` to see more rows

182

. 1

### dplyr::summarize

- Creates a new data frame.
- It returns one row for each combination of grouping variables

Example with no grouping variable:

>starwars.cut %>%

mutate(height.feet=height/30.48) %>%
summarize(mean=mean(height.feet,na.rm=TRUE),
median=median(height.feet,na.rm=TRUE))

### dplyr::summarize

Some basic statistics you may compute:

Objective	Function
Basic	mean()
	median()
	sum()
variation	sd()
	IQR()
range	min(), max()
	quantile()

#### summarize

Compute the average height in feet by gender

> starwars.cut %>%

mutate(height.feet=height/30.48) %>%
group\_by(gender) %>%
summarize(mean=mean(height.feet,na.rm=TRUE),
count=n())

# A tibble: 3 × 3

mean count

<db1> <int>

17

66

4

5.40

5.95

5.79

gender

 $\langle chr \rangle$ 

3 NA

feminine

masculine

Compute the deviation of each individual height (in feet) from the average for their gender

> starwars.cut %>%

mutate(height.feet=height/30.48) %>%

group\_by(gender) %>%

mutate(height.grp.mean=mean(height.feet,na.rm=TRUE)) %>%

mutate(height.dev=height-height.grp.mean) %>%

select(name,gender,height.feet, height.dev, height.grp.mean,)

# A tibble: 87 × 5				
# Groups: gender [3]				
name	gender	height.feet	height.dev	height.grp.mean
<chr></chr>	<chr></chr>	<db7></db7>	<db7></db7>	<db7></db7>
1 Luke Skywalker	masculine	5.64	-0.148	5.79
2 C-3PO	masculine	5.48	-0.312	5.79
3 R2-D2	masculine	3.15	-2.64	5.79
4 Darth Vader	masculine	6.63	0.836	5.79
5 Leia Organa	feminine	4.92	-0.482	5.40
6 Owen Lars	masculine	5.84	0.048 <u>7</u>	5.79
7 Beru Whitesun lars	feminine	5.41	0.010 <u>3</u>	5.40
8 R5-D4	masculine	3.18	-2.61	5.79
9 Biggs Darklighter	masculine	6.00	0.213	5.79
10 Obi-Wan Kenobi	masculine	5.97	0.180	5.79
# i 77 more rows				

# i Use `print(n = ...)` to see more rows

Also compute the deviation from the overall mean:

#### > starwars.cut %>%

- mutate(height.feet=height/30.48) %>%
- group\_by(gender) %>%
- mutate(height.dev=height.feet mean(height.feet,na.rm=TRUE))%>%
- mutate(height.grp.mean=mean(height.feet,na.rm=TRUE)) %>%
- ungroup(gender) %>%
- mutate(height.overall.dev=height.feet mean(height.feet,na.rm=TRUE)) %>%
  select(name,gender,height.feet,height.dev,height.grp.mean,height.overall.dev)

# A tibble: 87  $\times$  6

# /	A tibble: 87 × 6 🧖					
	name	gender	height.feet	height.dev	height.grp.mean	height.overall.dev
	<chr></chr>	<chr></chr>	<db1></db1>	<db1></db1>	<db1></db1>	<db7></db7>
1	Luke Skywalker	masculine	5.64	-0.148	5.79	-0.077 <u>4</u>
2	C-3PO	masculine	5.48	-0.312	5.79	-0.241
3	R2-D2	masculine	3.15	-2.64	5.79	-2.57
4	Darth Vader	masculine	6.63	0.836	5.79	0.907
5	Leia Organa	feminine	4.92	-0.482	5.40	-0.799
6	Owen Lars	masculine	5.84	0.048 <u>7</u>	5.79	0.119
- 7	Beru Whitesun lars	feminine	5.41	0.010 <u>3</u>	5.40	-0.307
8	R5-D4	masculine	3.18	-2.61	5.79	-2.54
9	Biggs Darklighter	masculine	6.00	0.213	5.79	0.284
10	Obi-Wan Kenobi	masculine	5.97	0.180	5.79	0.251

# i 77 more rows

# i Use `print(n = ...)` to see more rows

Output the count for each gender and eye color.

	gender	eye_color	count
	<chr></chr>	<chr></chr>	<int></int>
1	feminine	black	2
2	feminine	blue	6
3	feminine	brown	5
4	feminine	hazel	2
5	feminine	red, blue	1
6	feminine	yellow	1
7	masculine	black	8
8	masculine	blue	12
9	masculine	blue-gray	1
LO	masculine	brown	15
<b>i</b> #	13 more r	OWS	

Output the count for each gender and eye color.

> starwars.cut %>%

group\_by(gender, eye\_color) %>%
summarize(count=n())

	genaer	eye_color	count
	<chr></chr>	<chr></chr>	<int></int>
1	feminine	black	2
2	feminine	blue	6
3	feminine	brown	5
4	feminine	hazel	2
5	feminine	red, blue	1
6	feminine	yellow	1
7	masculine	black	8
8	masculine	blue	12
9	masculine	blue-gray	1
10	masculine	brown	15
# i	13 more r	OWS	

Select all gender="feminine" individuals with height in feet greater than the overall group average height and then summarize the count in each homeworld for this subgroup

# A tibble: 11 × 12

name	height	mass	hair_color	skin_color	eye_color	birth_year	sex	gender	homeworld	height.feet	height.mean
<chr></chr>	<int></int>	<db1></db1>	<chr></chr>	<chr></chr>	<chr></chr>	<db1></db1>	<chr></chr>	<chr></chr>	<chr></chr>	<db1></db1>	<db7></db7>
1 Beru White…	165	75	brown	light	blue	47	fema	femin	Tatooine	5.41	5.40
2 Ayla Secura	17 <b>8</b>	55	none	blue	hazel	48	fema	femin	Ryloth	5.84	5.40
3 Adi Gallia	184	50	none	dark	blue	NA	fema	femin	Coruscant	6.04	5.40
4 Luminara U.	. 170	56.2	black	yellow	blue	58	fema	femin	Mirial	5.58	5.40
5 Barriss Of…	. 166	50	black	yellow	blue	40	fema	femin	Mirial	5.45	5.40
6 Dormé	165	NA	brown	light	brown	NA	fema	femin	Naboo	5.41	5.40
7 Zam Wesell	168	55	blonde	fair, gre…	yellow	NA	fema	femin	Zolan	5.51	5.40
8 Taun We	213	NA	none	grey	black	NA	fema	femin	Kamino	6.99	5.40
9 Jocasta Nu	167	NA	white	fair	blue	NA	fema	femin…	Coruscant	5.48	5.40
10 Shaak Ti	178	57	none	red, blue…	black	NA	fema	femin	Shili	5.84	5.40
11 Padmé Amid.	165	45	brown	light	brown	46	fema	femin	Naboo	5.41	5.40

 $\sim$ 

# A tibble:  $8 \times 2$ homeworld count <chr> <int> 1 Coruscant 2 2 Kamino 1 3 Mirial 2 4 Naboo 2 5 Ryloth 1 6 Shili 1 7 Tatooine 1

8 Zolan 1

Select all gender="feminine" individuals with height in feet greater than the overall group average height and present a table summarizing the count of each homeworld in this subset

#### >starwars.cut %>%

filter(gender== 'feminine') %>%
mutate(height.feet=height/30.48) %>%
mutate(height.mean=mean(height.feet,na.rm=TRUE)) %>%

filter(height.feet > height.mean,na.rm=TRUE))

## Merging data frames

## dplyr::inner\_join

Only keeps observations from data frame A if they have a match in in data frame B. Unmatched observations from A or B are not kept

Example:

				Name	Wage/hr
	Name	Age	B=	Joe	15
A =	Joe	19		John	21
	John	33		Jack	24
	Jack	41		Ed	35

- > inner\_join(A,B,by='Name')
  Name Age Wage
- 1 Joe 19 15
- 2 John 33 21
- 3 Jack 41 24

## dplyr::left\_join and dplyr::right\_join

<u>^ _</u>	Name	Age
A =	Joe	19
	John	33
	Jack	41

_	Name	Wage/hr
B=	Joe	15
	John	21
	Jack	24
	Ed	35

>	left_	joir	n(A,B,	by='Name')
	Name	Age	Wage	
1	Joe	19	15	
2	John	33	21	
3	Jack	41	24	
>	right	_jo	in(A,B	,by='Name')
	Name	Age	Wage	
1	Joe	19	15	
2	John	33	21	
3	Jack	41	24	
4	Ed	NA	35	

## Helpful online communities

#### <u>R Studio</u>

• <u>https://forum.posit.co/</u>

#### <u>R</u>

<u>https://stackoverflow.com/questions/tagged/R</u>

#### <u>Tidyverse:</u>

- <u>https://posit.co/resources/videos/a-gentle-introduction-to-tidy-statistics-in-r/</u>
- <u>https://dplyr.tidyverse.org/reference/mutate-joins.html</u>

### **BERD House**

#### https://www.einsteinmed.edu/centers/ictr/biostatisticsepidemiology-research-design-core/berd-house/



The Division of Biostatistics offers biostatistics consulting and collaboration to enhance the quality and rigor of scientific research conducted by investigators at Einstein and Montefiore.

#### **Biostatistics walk-in clinic:**

Meet with a biostatistician without an appointment and obtain quick advice on your project.

X