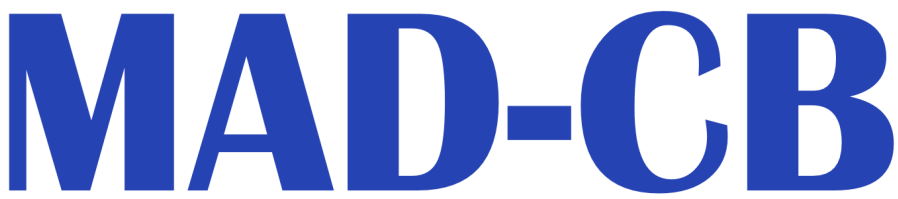


Figure 1:

Preparar Dados - Tidyverse

Preparar Dados para a Análise

Coleção dos Dados

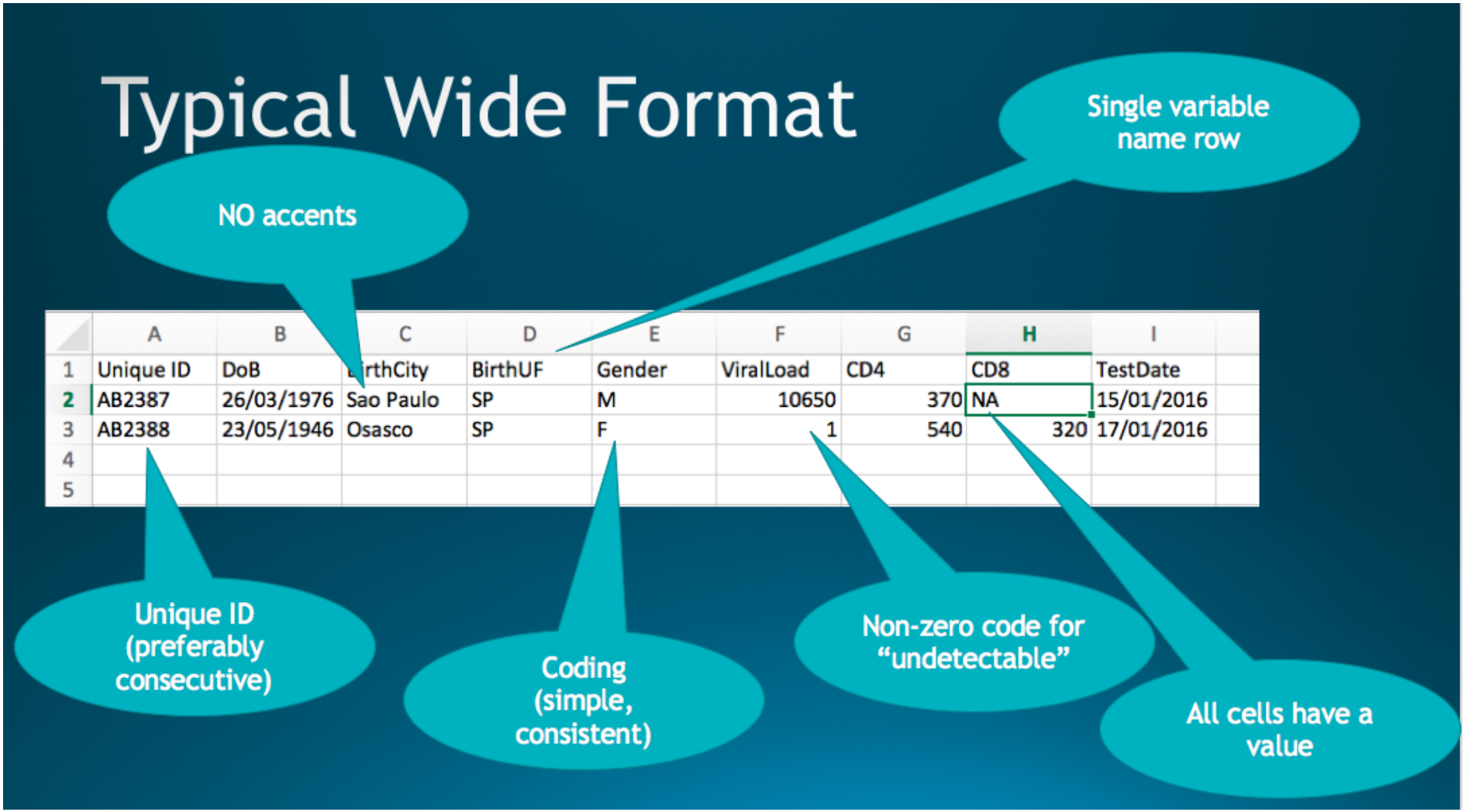
- Coletar os dados numa forma organizada do início
- Quais tipos de variáveis você quer
- Quais tipos de números eles são?
  - Categórico/Numérico?
- Fonte de dados?
  - Sondagens de opinião
  - Bases de dados existentes
  - Dados pessoais
  - Maquina (Sequenciador, PCR, etc.)

Onde Gravar os Dados

- Mais fácil: Excel ou equivalente
- Software de base de dados (SQL, outro)

Formato para Gravar os Dados

- Formato “wide”
  - Cada linha é uma observação completa
  - Variáveis ficam nas colunas
  - De preferência, 1ª coluna é um identificador único
  - Um ID que liga o fonte de observações através de um serie de tabelas
  - Ex: RENAGENO: Patient ID
- Formato “wide” facilita transferência dos dados da planilha a um software de análise



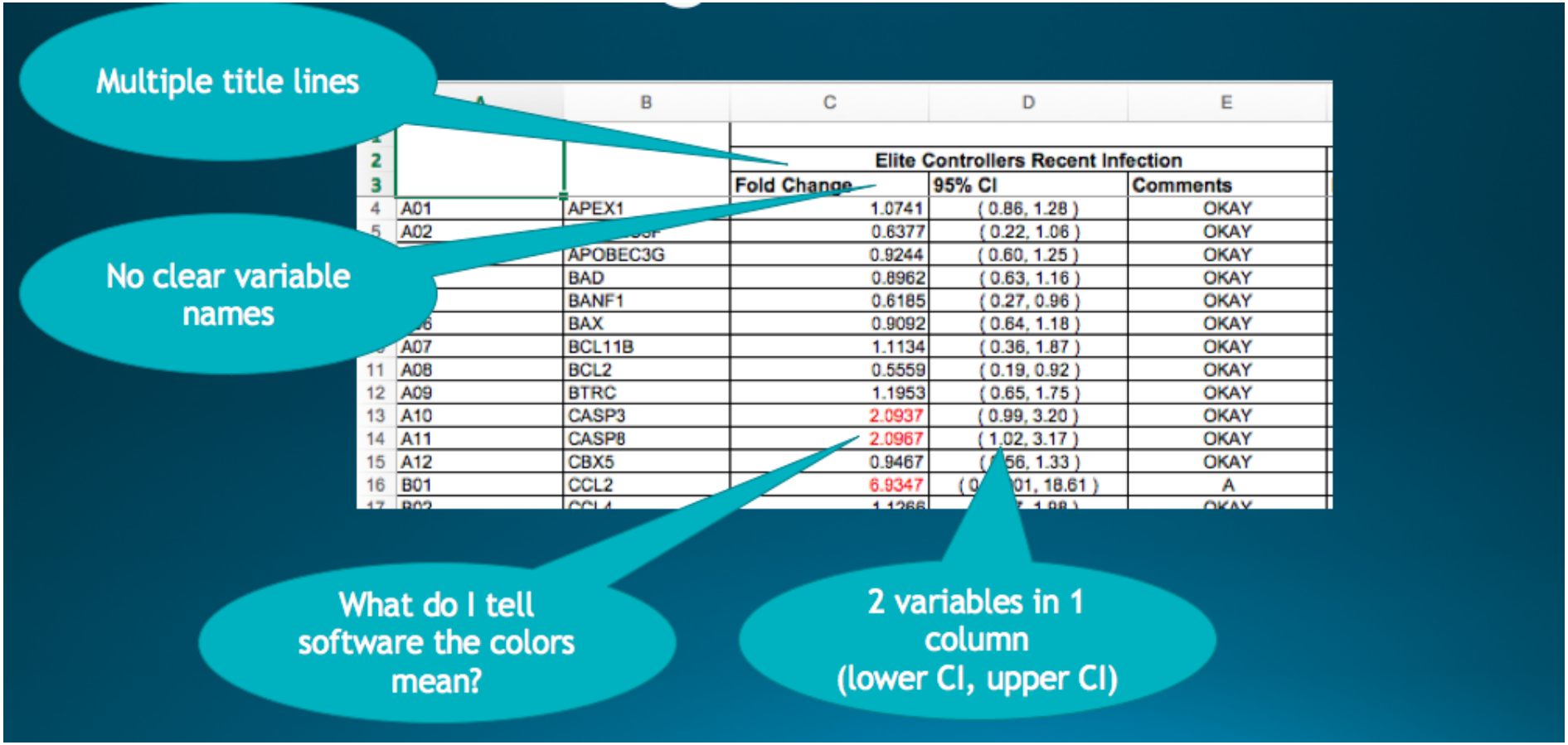
	A	B	C	D	E	F	G	H	I
1	Unique ID	DoB	BirthCity	BirthUF	Gender	ViralLoad	CD4	CD8	TestDate
2	AB2387	26/03/1976	Sao Paulo	SP	M	10650	370	NA	15/01/2016
3	AB2388	23/05/1946	Osasco	SP	F	1	540	320	17/01/2016
4									
5									

Figure 2:

Pontos Chaves

- Toda linha — **observação**
- Toda coluna — **variável**

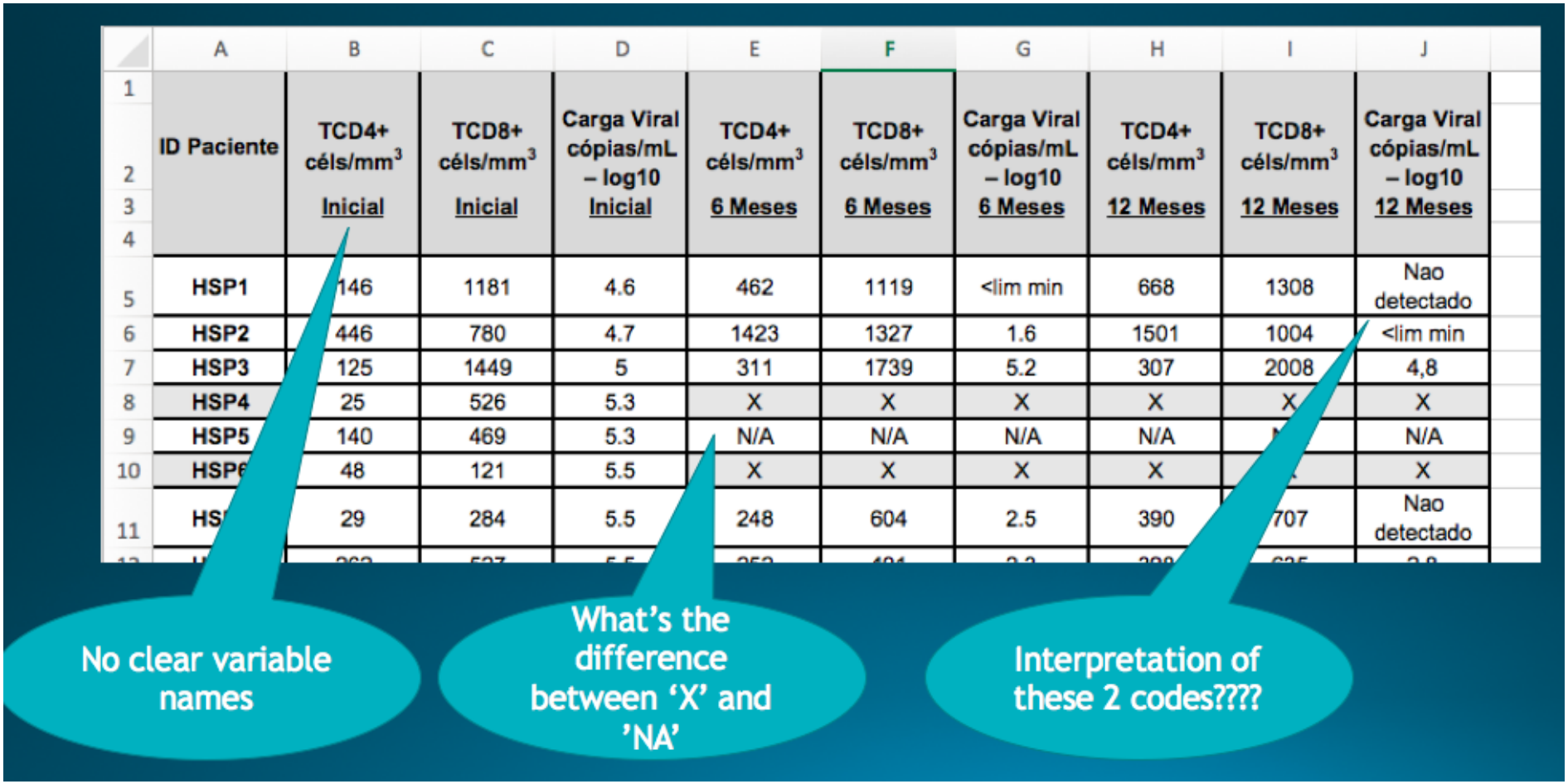
Um Exemplo Não Tão Bom



	A	B	C	D	E
2			Elite Controllers Recent Infection		
3			Fold Change	95% CI	Comments
4	A01	APEX1	1.0741	(0.86, 1.28)	OKAY
5	A02		0.6377	(0.22, 1.06)	OKAY
		APOBEC3G	0.9244	(0.60, 1.25)	OKAY
		BAD	0.8962	(0.63, 1.16)	OKAY
		BANF1	0.6185	(0.27, 0.96)	OKAY
		BAX	0.9092	(0.64, 1.18)	OKAY
	A07	BCL11B	1.1134	(0.36, 1.87)	OKAY
11	A08	BCL2	0.5559	(0.19, 0.92)	OKAY
12	A09	BTRC	1.1953	(0.65, 1.75)	OKAY
13	A10	CASP3	2.0937	(0.99, 3.20)	OKAY
14	A11	CASP8	2.0967	(1.02, 3.17)	OKAY
15	A12	CBX5	0.9467	(56, 1.33)	OKAY
16	B01	CCL2	6.9347	(01, 18.61)	A

Figure 3:

E Essa Planilha



	A	B	C	D	E	F	G	H	I	J
1										
2	ID Paciente	TCD4+ céls/mm^3 Inicial	TCD8+ céls/mm^3 Inicial	Carga Viral cópias/mL – log10 Inicial	TCD4+ céls/mm^3 6 Meses	TCD8+ céls/mm^3 6 Meses	Carga Viral cópias/mL – log10 6 Meses	TCD4+ céls/mm^3 12 Meses	TCD8+ céls/mm^3 12 Meses	Carga Viral cópias/mL – log10 12 Meses
3										
4										
5	HSP1	146	1181	4.6	462	1119	<lim min	668	1308	Nao detectado
6	HSP2	446	780	4.7	1423	1327	1.6	1501	1004	<lim min
7	HSP3	125	1449	5	311	1739	5.2	307	2008	4,8
8	HSP4	25	526	5.3	X	X	X	X	X	X
9	HSP5	140	469	5.3	N/A	N/A	N/A	N/A	N	N/A
10	HSP6	48	121	5.5	X	X	X	X	X	X
11	HS	29	284	5.5	248	604	2.5	390	707	Nao detectado

No clear variable names

What's the difference between 'X' and 'NA'

Interpretation of these 2 codes????

Figure 4:

Resumo – Coleção de Dados/Gravação

- Objetivo – preparar os dados para análise
  - Não publicação ou divulgação
  - Ênfase – fazer os dados compreensíveis para o software analítico
- Planilha com Dados Precisa
  - Dicionário dos Dados
  - Listagem de todas as variáveis, significação, e códigos usados
  - Você não vai lembrar os códigos depois de um ano sem um "cheat sheet"
- **SEJA PRECISO; NÃO CHUTE!**
  - A vida e saúde das pessoas podem depender em seus resultados
  - Arredondar números nesta fase não permitido
- "NA" é o código certo para "Não tenho este número"
  - Não 0 (zero) – Zero é um número que quer dizer algo especifico

Exemplo de Uma Planilha Razoável

	A	B	C	D	E	F	G	H	I	J	K	L
1	codepac	idade	sexo	cidnasc	ufnasc	raca	escol	gestante	amostraid	copias_cv	contagem_cd4	contagem_cd8
2	96710	60	Masculino	Torres	RS	NA	NA	Nao	05RS090077	5200	898	1311
3	93778	73	Masculino	Vargem Grande do Sul	SP	Branca	De 8 a 11 anc	Nao	11SP073735	1947	958	817
4	91200	51	Feminino	Rinopolis	SP	Parda	De 4 a 7 anos	Nao	11SP073769	480000	958	817
5	91228	50	Masculino	Porto Feliz	SP	Branca	De 8 a 11 anc	NA	13SP070671	257313	142	1009
6	96186	44	Feminino	Ico	CE	Parda	De 4 a 7 anos	Nao	11SP073423	2585	524	586
7	93513	63	Masculino	Barreirinhas	MA	NA	NA	Nao	04DF080006	84	256	651
8	94147	25	Masculino	Lins	SP	Parda	De 4 a 7 anos	Nao	19SP090306	1286	353	393
9	99352	61	Feminino	Santana do Livramento	RS	NA	NA	Nao	21PR080198	13000	928	1740

Figure 5:

Dicionário dos Dados

	A	B	C	D	E	F
1	Nome	Tipo	Descreve			
2	codepac	Integer/Cat	ID de paciente			
3	idade	Integer	Idade em anos			
4	sexo	Cat	Genero (Masculino ou Feminino)			
5	cidnasc	Cat	Cidade de Nascimento			
6	ufnasc	Cat	Estado de Nascimento			
7	raca	Cat	Cor de pele			
8	escol	Cat	Número de anos de escolaridade			
9	gestante	Lógico	Se gestante ou não			
10	amostraid	Cat	Identificador de amostra de genotipagem			
11	copias_cv	Numérico	Número de cópias do vírus			
12	contagem_cd4	Numérico	Número de celulas T com CD4+/ml			
13	contagem_cd8	Numérico	Número de celulas T com CD8+/ml			
14						

Nossos Dados “Tidy”?

- Variáveis em colunas; Observações (pacientes) em linhas (OK)
- Podemos trabalhar com esses dados

Organização de Dados - “Tidy Data”

“Tidy Data” = Dados Organizados

- Dados seguem um formato consistente
- Um mapeamento da significação do conjunto à estrutura dele
- Facilitar a localização dos elementos do conjunto
- Facilitar o cálculo de estatísticas e construção dos gráficos
- Facilitar a percepção das relações entre variáveis

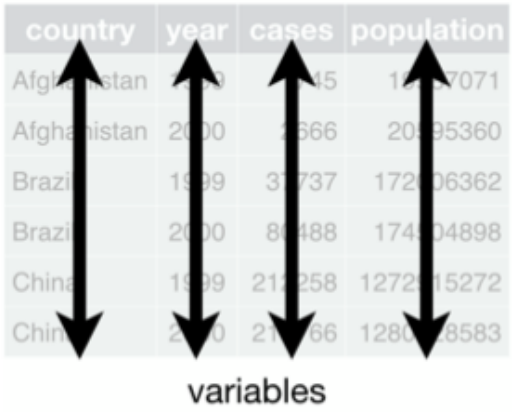
Definição de 'Tidy Data' de Hadley Wickham

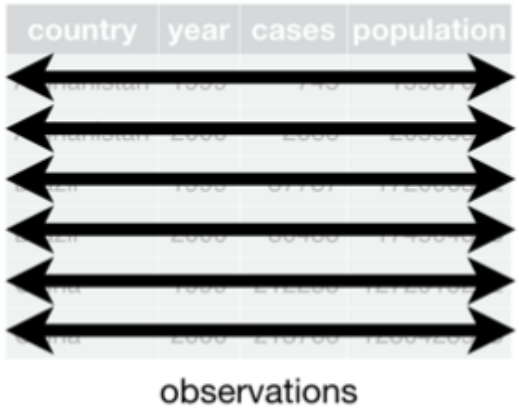
> *A dataset is a collection of values, usually either* ***numbers*** *(if quantitative) or* ***strings*** *(if qualitative). Values are organised in two ways.* ***Every value belongs to a variable and an observation.*** *A variable contains all values that measure the same underlying attribute (like height, temperature, duration) across units. An observation contains all values measured on the same unit (like a person, or a day, or a race) across attributes.*

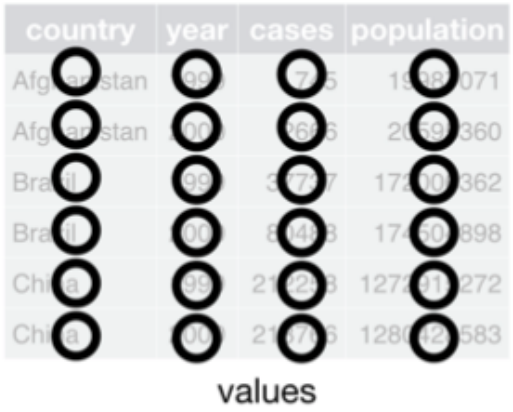
Wickham, Hadley. 2014. "Tidy Data." Journal of Statistical Software Volume 59 (Issue 10). `https://www.jstatsoft.org/index.php/jss/article/view/v059i10/v59i10.pdf`.

3 Características de Tidy Data

1. Cada variável fica numa coluna
2. Cada observação fica numa linha
3. Cada tipo de unidade observacional compõe uma tabela.







Por Esta Definição, Nossos Dados Tidy?

- Eles combinam vários tipos de dados no mesmo conjunto

	A	B	C	D	E	F	G	H	I	J	K	L
1	codepac	idade	sexo	cidnasc	ufnasc	raca	escol	gestante	amostraid	copias_cv	contagem_cd4	contagem_cd8
2	96710	60	Masculino	Torres	RS	NA	NA	Nao	05RS090077	5200	898	1311
3	93778	73	Masculino	Vargem Grande do Sul	SP	Branca	De 8 a 11 anc	Nao	11SP073735	1947	958	817
4	91200	51	Feminino	Rinopolis	SP	Parda	De 4 a 7 anos	Nao	11SP073769	480000	958	817
5	91228	50	Masculino	Porto Feliz	SP	Branca	De 8 a 11 anc	NA	13SP070671	257313	142	1009
6	96186	44	Feminino	Ico	CE	Parda	De 4 a 7 anos	Nao	11SP073423	2585	524	586
7	93513	63	Masculino	Barreirinhas	MA	NA	NA	Nao	04DF080006	84	256	651
8	94147	25	Masculino	Lins	SP	Parda	De 4 a 7 anos	Nao	19SP090306	1286	353	393
9	99352	61	Feminino	Santana do Livramento	RS	NA	NA	Nao	21PR080198	13000	928	1740

- Algumas variáveis são informações demográficas dos pacientes
  - `idade`, `sexo`, `cidnasc`
- Outras contam resultados quantitativos dos testes
  - `copies_cv`, `contagem_cd4`, `contagem_cd8`
- O que unificam os 2 tipos é o `codepac` – o ID do paciente

Podemos fazer o trabalho de separar esses tipos de dados em R

Passo 1 – Chamar os Pacotes Que Usaremos

```
suppressMessages(library(tidyverse))
library(DescTools)
```

Passo 2 – Carregar os Dados e Olhar Neles

```
dados <- read_csv("pac_demo.csv")
```

```
## Parsed with column specification:
## cols(
##   codepac = col_integer(),
##   idade = col_integer(),
##   sexo = col_character(),
##   cidnasc = col_character(),
##   ufnasc = col_character(),
##   raca = col_character(),
##   escol = col_character(),
##   gestante = col_character(),
##   amostraid = col_character(),
##   copias_cv = col_integer(),
##   contagem_cd4 = col_integer(),
##   contagem_cd8 = col_integer()
## )
```

```r
tibble::glimpse(dados)
```

```
## Observations: 50
## Variables: 12
## $ codepac      <int> 96710, 93778, 91200, 91228, 96186, 93513, 94147, ..
## $ idade        <int> 60, 73, 51, 50, 44, 63, 25, 61, 49, 41, 44, 81, 2..
## $ sexo         <chr> "Masculino", "Masculino", "Feminino", "Masculino"..
## $ cidnasc      <chr> "Torres", "Vargem Grande do Sul", "Rinopolis", "P..
## $ ufnasc       <chr> "RS", "SP", "SP", "SP", "CE", "MA", "SP", "RS", "..
## $ raca         <chr> NA, "Branca", "Parda", "Branca", "Parda", NA, "Pa..
## $ escol        <chr> NA, "De 8 a 11 anos", "De 4 a 7 anos", "De 8 a 11..
## $ gestante     <chr> "Nao", "Nao", "Nao", NA, "Nao", "Nao", "Nao", "Na..
## $ amostraid    <chr> "05RS090077", "11SP073735", "11SP073769", "13SP07..
## $ copias_cv    <int> 5200, 1947, 480000, 257313, 2585, 84, 1286, 13000..
## $ contagem_cd4 <int> 898, 958, 958, 142, 524, 256, 353, 928, 66, 66, 3..
## $ contagem_cd8 <int> 1311, 817, 817, 1009, 586, 651, 393, 1740, 801, 8..
```

Passo 3 – Fazer o Conjunto Realmente "Tidy" por Subsets

- Subsetting por `dplyr::select`
  - Criar um subset só com os dados demográficos – `demog`
  - Criar um outro subset só com os dados de testes – `testes`
  - O conjunto mestre fica o mesmo.
  - Sempre temos isso para referência e criação de novos subsets

Extract Variables

Column functions return a set of columns as a new table. Use a variant that ends in _ for non-standard evaluation friendly code.

select(.data, ...)

Extract columns by name. Also **select_if()**

select(iris, Sepal.Length, Species)

Use these helpers with select(),

e.g. *select(iris, starts_with("Sepal"))*

contains(match)	**num_range**(prefix, range)	**:**, e.g. mpg:cyl
ends_with(match)	**one_of**(...)	**-**, e.g, -Species
matches(match)	**starts_with**(match)	

```
demog <- dados %>% select(codepac:amostraid)
testes <- dados %>% select(c(codepac, copias_cv:contagem_cd8))
```

demog Dados

```
glimpse(demog)
```

```
## Observations: 50
## Variables: 9
## $ codepac   <int> 96710, 93778, 91200, 91228, 96186, 93513, 94147, 993..
## $ idade     <int> 60, 73, 51, 50, 44, 63, 25, 61, 49, 41, 44, 81, 25, ..
## $ sexo      <chr> "Masculino", "Masculino", "Feminino", "Masculino", "..
## $ cidnasc   <chr> "Torres", "Vargem Grande do Sul", "Rinopolis", "Port..
## $ ufnasc    <chr> "RS", "SP", "SP", "SP", "CE", "MA", "SP", "RS", "RS"..
## $ raca      <chr> NA, "Branca", "Parda", "Branca", "Parda", NA, "Parda..
## $ escol     <chr> NA, "De 8 a 11 anos", "De 4 a 7 anos", "De 8 a 11 an..
## $ gestante  <chr> "Nao", "Nao", "Nao", NA, "Nao", "Nao", "Nao", "Nao",..
## $ amostraid <chr> "05RS090077", "11SP073735", "11SP073769", "13SP07067..
```

Variáveis com Número Limitado de Categorias

- `sexo` - “Masculino”, “Feminino” – 2 (fácil)
- Para `raca`, `escol`, `gestante`, pode usar a função `unique` para ver quantas categorias
- `unique`: retorna os elementos únicos de uma variável

```
unique(demog$raca)
```

```
## [1] NA       "Branca" "Parda"  "Preta"
```

```
unique(demog$escol)
```

```
## [1] NA                  "De 8 a 11 anos"    "De 4 a 7 anos"
## [4] "De 1 a 3 anos"     "De 12 e mais anos" "Nenhuma"
```

```
unique(demog$gestante)
```

```
## [1] "Nao" NA
```

Passo 4 – Simplificar as Variáveis Categoricas

- Converter todos os 4 ao tipo de `factor`
  - Não é necessário com outros porque têm muitas categorias
- `gestante` - só tem um valor; não necessário para os cálculos; retirar ela
- `raca` - categorias servem; compreensíveis
- `escol` - usamos "fundamental", "média", "superior" no dia-à-dia
  - Simplificar as atuais

`factor` como Classe de Variável

- Variáveis categóricas têm número de nivéis fixos e conhecidos
  - Ex., “Masculino”/“Feminino”
- `factor` é uma classe que gerencia elas com eficiência
- `factor` converte categóricas em números internamente mas deixa o valor original como `character`
- Com `factor` pode controle a ordem das categorias

Exemplo: `factor` – Meses do Ano

- Imagine a variável mes que tem 4 valores: “jan”, “mai”, “out”, “abr”

```
(meses <- c("jan", "mai", "out", "abr"))
```

```
## [1] "jan" "mai" "out" "abr"
```

```
sort(meses) ## colocar os valores em ordem
```

```
## [1] "abr" "jan" "mai" "out"
```

Quisemos eles em ordem de mês. Como Podemos Fazer?

1. Ensinar R o que são os nivéis possíveis – os meses
2. Converter `meses` para um `factor`

```
mes_level <- c("jan", "fev", "mar", "abr", "mai", "jun", "jul", "ago",
               "set", "out", "nov", "dez")
mesesf <- factor(meses, levels = mes_level)
str(mesesf)
```

```
##  Factor w/ 12 levels "jan","fev","mar",..: 1 5 10 4
```

```
sort(mesesf)
```

```
## [1] jan abr mai out
## Levels: jan fev mar abr mai jun jul ago set out nov dez
```

Passo 4A – Criar Fatores em demog

```
demog <- demog %>%
         mutate(sexo = factor(sexo, levels = c("Masculino", "Feminino"))) %>%
         mutate(raca = factor(raca, levels = c("Branca", "Parda", "Preta"))) %>%
         mutate(escol = factor(escol)) # mudança de valores mais tarde
```

Agora, com Factors, Fácil a Contar Variáveis Categóricas

```
demog %>% dplyr::count(sexo)
```

```
## # A tibble: 2 × 2
##        sexo     n
##      <fctr> <int>
## 1 Masculino    28
## 2  Feminino    22
```

Função para Mudar Valores de Factores – `forcats::fct_recode()`

- `fct_recode` muda os valores de factors seguindo seu comando
- `fct_recode(x, <valor novo> = <valor velho> , ...)`
- Aplicado à `sexo`: 'fct_recode(demog$sexo, m = "Masculino")

```
library(forcats)  ## Carregar "forcats"; instalar se for necessário
demog <- demog %>%
         mutate(sexo = fct_recode(sexo,
                                  "mas" = "Masculino",
                                  "fem" = "Feminino"))
str(demog$sexo)
```

```
##  Factor w/ 2 levels "mas","fem": 1 1 2 1 2 1 1 2 2 2 ...
```

Para Mudar `escol`

- Queremos a categoria “fundamental” para tratar de 7 ou menos anos
  - “De 1 a 3 anos”
  - “De 4 a 7 anos”
- Queremos a categoria “media” para 8 - 11 anos
  - “De 8 a 11 anos”
- Queremos a categoria “superior” para 12 ou mais anos
  - “De 12 e mais anos”
- “Nenhuma” só queremos mudar para minísculo “nenhuma”
- Podemos dar um de novos nomes às múltiplas categorias velhas

```r
demog <- demog %>%
         mutate(escol = fct_recode(escol,
                                   "fundamental" = "De 1 a 3 anos",
                                   "fundamental" = "De 4 a 7 anos",
                                   "media" = "De 8 a 11 anos",
                                   "superior" = "De 12 e mais anos",
                                   "nenhuma" = "Nenhuma"))
str(demog$escol)
```

```
##  Factor w/ 4 levels "fundamental",..: NA 3 1 3 1 NA 1 NA NA NA ...
```

```r
levels(demog$escol)
```

```
## [1] "fundamental" "superior"    "media"       "nenhuma"
```