(https://github.com/dd-consulting)

One-Stop Analytics: Exploratory Data Analysis (EDA) & Statistics

Case Study of Autism Spectrum Disorder (ASD) with R

[United States]

Centers for Disease Control and Prevention (CDC) - Autism Spectrum Disorder (ASD)

Autism spectrum disorder (ASD) is a developmental disability that can cause significant social, communication and behavioral challenges. CDC is committed to continuing to provide essential data on ASD, search for factors that put children at risk for ASD and possible causes, and develop resources that help identify children with ASD as early as possible.

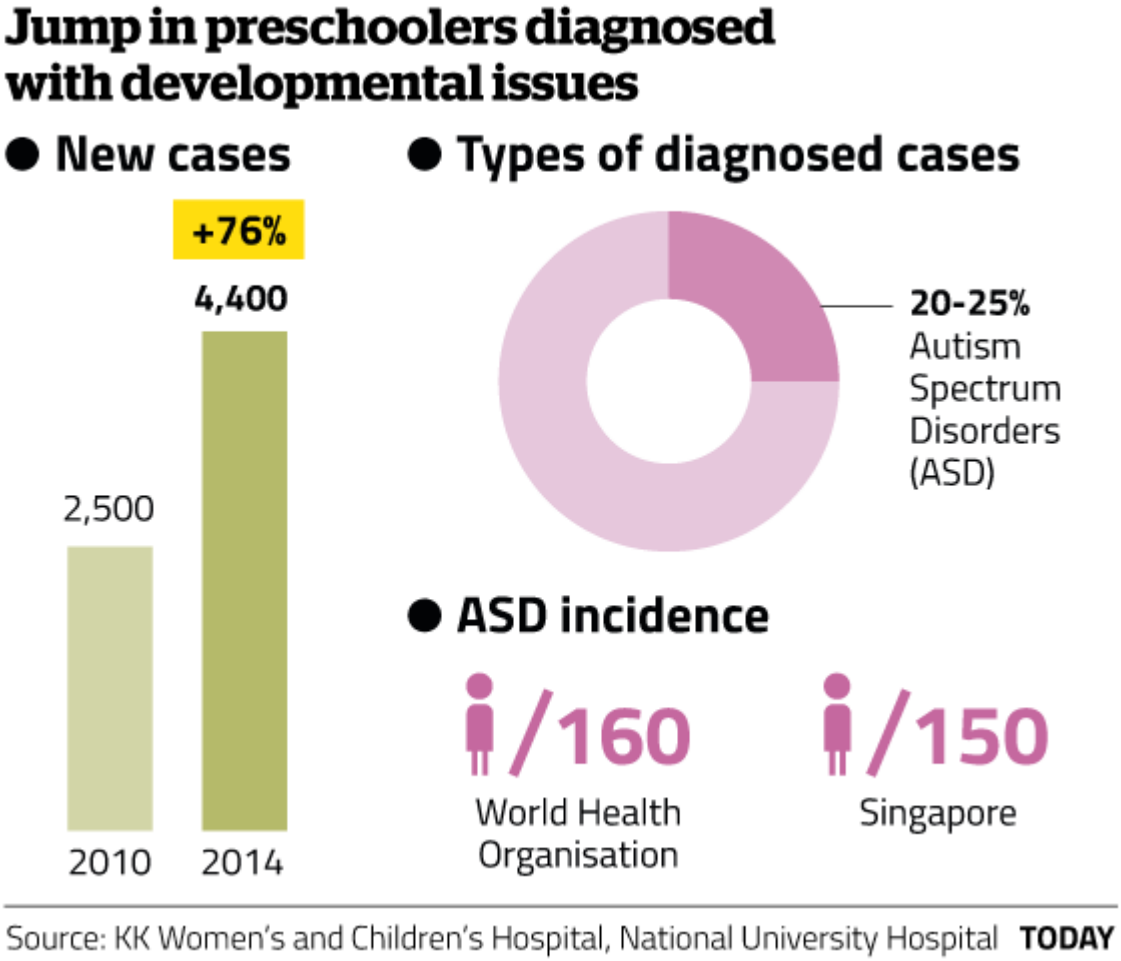
https://www.cdc.gov/ncbddd/autism/data/index.html (https://www.cdc.gov/ncbddd/autism/data/index.html)

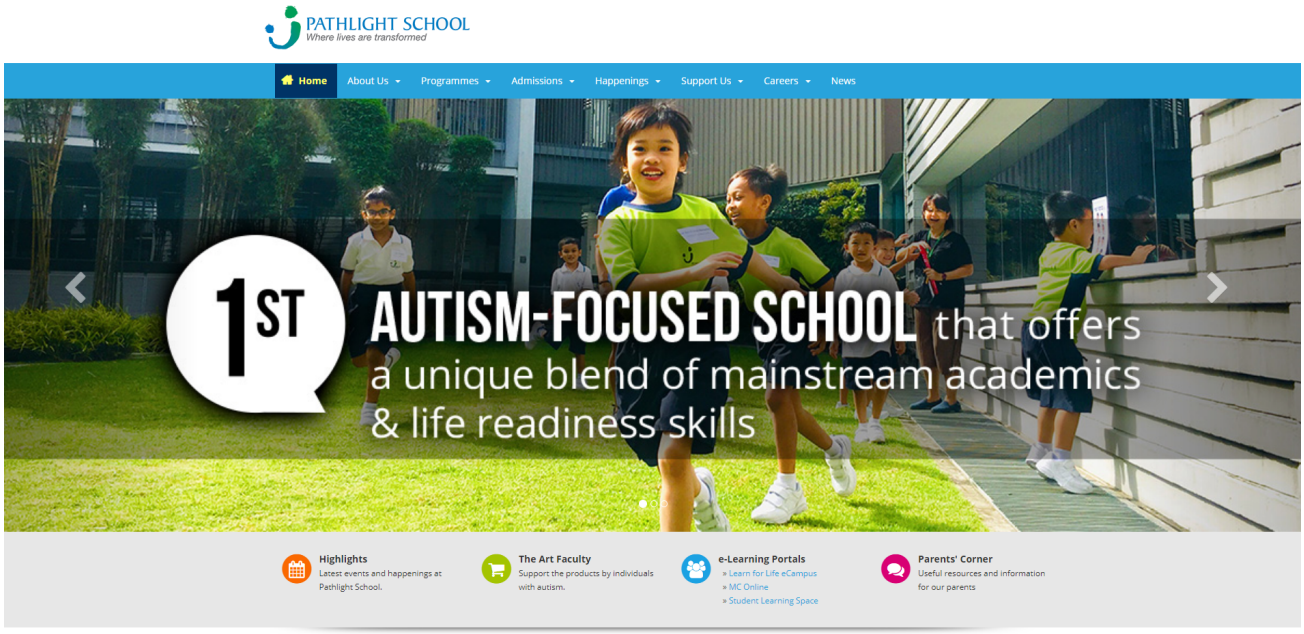
[Singapore]

TODAY Online - More preschoolers diagnosed with developmental issues

Doctors cited better awareness among parents and preschool teachers, leading to early referrals for diagnosis.

https://www.gov.sg/news/content/today-online-more-preschoolers-diagnosed-with-developmental-issues (https://www.gov.sg/news/content/today-online-more-preschoolers-diagnosed-with-developmental-issues)





https://www.pathlight.org.sg/ (https://www.pathlight.org.sg/)

()

Workshop Objective:

Use R to analyze Autism Spectrum Disorder (ASD) data from CDC USA.

https://www.cdc.gov/ncbddd/autism/data/index.html (https://www.cdc.gov/ncbddd/autism/data/index.html)

- R Fundamentals
- Data Summarization
- Data Visualisation (Base Graphic)
- Data Visualisation (Enhanced)
- Sampling & Normality
- Confidence Interval (CI)
- Workshop Submission
- Appendices

()

R Fundamentals

R Fundamentals - Get & Set working directory

Obtain current R working directory

In [1]:
```
getwd()
```

'/media/sf_vm_shared_folder/git/DDC/DDC-ASD/model_R'

Set new R working directory

In [2]:
```
# setwd("/media/sf_vm_shared_folder/git/DDC/DDC-ASD/model_R")
# setwd('~/Desktop/admin-desktop/vm_shared_folder/git/DDC-ASD/model_R')
getwd()
```

'/media/sf_vm_shared_folder/git/DDC/DDC-ASD/model_R'

Read in CSV data, storing as R dataframe

In [3]:
```
# Dataset: US. National Level Children ASD Prevalence
ASD_National <- read.csv("../dataset/ADV_ASD_National.csv", stringsAsFactors =
```

In [4]:
```
# Dataset: US. State Level Children ASD Prevalence
ASD_State    <- read.csv("../dataset/ADV_ASD_State.csv", stringsAsFactors = FA
```

Look at first/last few rows of data

In [5]:
```
head(ASD_National)
```

Source	Year	Prevalence	Upper.CI	Lower.CI	Prevalence_dup	Source_Full1	Source_Full2	Male.Preval
addm	2000	6.7	7.0	6.3	6.7	Autism & Developmental Disabilities Monitoring Network	addm-Autism & Developmental Disabilities Monitoring Network	Nc
addm	2002	6.6	6.8	6.3	6.6	Autism & Developmental Disabilities Monitoring Network	addm-Autism & Developmental Disabilities Monitoring Network	1
addm	2004	8.0	8.4	7.6	8.0	Autism & Developmental Disabilities Monitoring Network	addm-Autism & Developmental Disabilities Monitoring Network	1
addm	2006	9.0	9.3	8.6	9.0	Autism & Developmental Disabilities Monitoring Network	addm-Autism & Developmental Disabilities Monitoring Network	1
addm	2008	11.3	11.7	11.0	11.3	Autism & Developmental Disabilities Monitoring Network	addm-Autism & Developmental Disabilities Monitoring Network	1
addm	2010	14.7	15.1	14.3	14.7	Autism & Developmental Disabilities Monitoring Network	addm-Autism & Developmental Disabilities Monitoring Network	2

In [6]:
```
tail(ASD_State)
```

	State	Denominator	Prevalence	Lower.CI	Upper.CI	Year	Source	Source_Full1	State_Full1	Stat
1687	UT	596257	8.7	8.5	9.0	2016	sped	Special Education Child Count	Utah	U
1688	VT	74108	12.1	11.3	12.9	2016	sped	Special Education Child Count	Vermont	VT-V
1689	VA	1162945	14.2	14.0	14.4	2016	sped	Special Education Child Count	Virginia	VA-
1690	WA	1006676	11.2	11.0	11.4	2016	sped	Special Education Child Count	Washington	Was
1691	WV	239037	8.6	8.3	9.0	2016	sped	Special Education Child Count	West Virginia	W
1692	WY	85922	9.3	8.7	10.0	2016	sped	Special Education Child Count	Wyoming	W

Obtain number of rows and number of columns/features/variables

In [7]:
```
dim(ASD_National)
```

```
42  26
```

In [8]:
```
dim(ASD_State)
```

```
1692  49
```

Obtain overview (data structure/types)

In [9]:
```
str(ASD_National)
```

```
'data.frame':   42 obs. of  26 variables:
 $ Source                              : chr  "addm" "addm" "addm" "addm"
...
 $ Year                                : int  2000 2002 2004 2006 2008 2010
2012 2014 2004 2008 ...
 $ Prevalence                          : num  6.7 6.6 8 9 11.3 14.7 14.8 1
6.8 9.5 16.2 ...
 $ Upper.CI                            : num  7 6.8 8.4 9.3 11.7 15.1 15.2
17.3 12 18.1 ...
 $ Lower.CI                            : num  6.3 6.3 7.6 8.6 11 14.3 14.4
16.4 7.4 14.5 ...
 $ Prevalence_dup                      : num  6.7 6.6 8 9 11.3 14.7 14.8 1
6.8 9.5 16.2 ...
 $ Source_Full1                        : chr  "Autism & Developmental Disab
ilities Monitoring Network" "Autism & Developmental Disabilities Monitoring
Network" "Autism & Developmental Disabilities Monitoring Network" "Autism &
Developmental Disabilities Monitoring Network" ...
 $ Source_Full2                        : chr  "addm-Autism & Developmental
Disabilities Monitoring Network" "addm-Autism & Developmental Disabilities
```

In [10]:
```
str(ASD_State)
```

```
'data.frame':   1692 obs. of  49 variables:
 $ State                                : chr  "AZ" "GA" "MD" "NJ" ...
 $ Denominator                          : int  45322 43593 21532 29714 245
35 23065 35472 45113 36472 11020 ...
 $ Prevalence                           : num  6.5 6.5 5.5 9.9 6.3 4.5 3.3
6.2 6.9 5.9 ...
 $ Lower.CI                             : num  5.8 5.8 4.6 8.9 5.4 3.7 2.7
5.5 6.1 4.6 ...
 $ Upper.CI                             : num  7.3 7.3 6.6 11.1 7.4 5.5 3.
9 7 7.8 7.5 ...
 $ Year                                 : int  2000 2000 2000 2000 2000 20
00 2002 2002 2002 2002 ...
 $ Source                               : chr  "addm" "addm" "addm" "addm"
...
 $ Source_Full1                         : chr  "Autism & Developmental Dis
abilities Monitoring Network" "Autism & Developmental Disabilities Monitori
ng Network" "Autism & Developmental Disabilities Monitoring Network" "Autis
m & Developmental Disabilities Monitoring Network" ...
 $ State_Full1                          : chr  "Arizona" "Georgia" "Maryla
nd" "New Jersey"
```

Obtain name of columns

In [11]:
```
names(ASD_National)
```

'Source' 'Year' 'Prevalence' 'Upper.CI' 'Lower.CI' 'Prevalence_dup' 'Source_Full1' 'Source_Full2' 'Male.Prevalence' 'Male.Lower.CI' 'Male.Upper.CI' 'Female.Prevalence' 'Female.Lower.CI' 'Female.Upper.CI' 'Non.hispanic.white.Prevalence' 'Non.hispanic.white.Lower.CI' 'Non.hispanic.white.Upper.CI' 'Non.hispanic.black.Prevalence' 'Non.hispanic.black.Lower.CI' 'Non.hispanic.black.Upper.CI' 'Hispanic.Prevalence' 'Hispanic.Lower.CI' 'Hispanic.Upper.CI' 'Asian.or.Pacific.Islander.Prevalence' 'Asian.or.Pacific.Islander.Lower.CI' 'Asian.or.Pacific.Islander.Upper.CI'

In [12]:
```
names(ASD_State)
```

'State' 'Denominator' 'Prevalence' 'Lower.CI' 'Upper.CI' 'Year' 'Source' 'Source_Full1' 'State_Full1' 'State_Full2' 'Numerator_ASD' 'Numerator_NonASD' 'Proportion' 'X95_Z_CI' 'Z_Lower.CI' 'Z_Upper.CI' 'Z_Lower.CI_ABSerror' 'Z_Upper.CI_ABSerror' 'Chi_Wilson_P' 'X95_Chi_Wilson_CI' 'Chi_Wilson_Lower.CI' 'Chi_Wilson_Upper.CI' 'Chi_Wilson_Lower.CI_ABSerror' 'Chi_Wilson_Upper.CI_ABSerror' 'Chi_Wilson_Corrected_w_minus.CI' 'Chi_Wilson_Corrected_w_plus.CI' 'Chi_Wilson_Corrected_Lower.CI' 'Chi_Wilson_Corrected_Upper.CI' 'Chi_Wilson_Corrected_Lower.CI_ABSerror' 'Chi_Wilson_Corrected_Upper.CI_ABSerror' 'Male.Prevalence' 'Male.Lower.CI' 'Male.Upper.CI' 'Female.Prevalence' 'Female.Lower.CI' 'Female.Upper.CI' 'Non.hispanic.white.Prevalence' 'Non.hispanic.white.Lower.CI' 'Non.hispanic.white.Upper.CI' 'Non.hispanic.black.Prevalence' 'Non.hispanic.black.Lower.CI' 'Non.hispanic.black.Upper.CI' 'Hispanic.Prevalence' 'Hispanic.Lower.CI' 'Hispanic.Upper.CI' 'Asian.or.Pacific.Islander.Prevalence' 'Asian.or.Pacific.Islander.Lower.CI' 'Asian.or.Pacific.Islander.Upper.CI' 'State_Region'

Display column name with its index number

In [13]:
```
cbind(names(ASD_National), c(1:length(names(ASD_National))))
```

Source	1
Year	2
Prevalence	3
Upper.CI	4
Lower.CI	5
Prevalence_dup	6
Source_Full1	7
Source_Full2	8
Male.Prevalence	9
Male.Lower.CI	10
Male.Upper.CI	11
Female.Prevalence	12
Female.Lower.CI	13

Look at data structure/schema (Selected columns)

In [14]:
```
str(ASD_National[, c(1:8, 24, 25, 26)])
```

```
'data.frame':   42 obs. of  11 variables:
 $ Source                                : chr  "addm" "addm" "addm" "addm" ...
 $ Year                                  : int  2000 2002 2004 2006 2008 2010 2
012 2014 2004 2008 ...
 $ Prevalence                            : num  6.7 6.6 8 9 11.3 14.7 14.8 16.8
9.5 16.2 ...
 $ Upper.CI                              : num  7 6.8 8.4 9.3 11.7 15.1 15.2 1
7.3 12 18.1 ...
 $ Lower.CI                              : num  6.3 6.3 7.6 8.6 11 14.3 14.4 1
6.4 7.4 14.5 ...
 $ Prevalence_dup                        : num  6.7 6.6 8 9 11.3 14.7 14.8 16.8
9.5 16.2 ...
 $ Source_Full1                          : chr  "Autism & Developmental Disabil
ities Monitoring Network" "Autism & Developmental Disabilities Monitoring Net
work" "Autism & Developmental Disabilities Monitoring Network" "Autism & Deve
lopmental Disabilities Monitoring Network" ...
 $ Source_Full2                          : chr  "addm-Autism & Developmental Di
sabilities Monitoring Network" "addm-Autism & Developmental Disabilities Moni
toring Network" "addm-Autism & Developmental Disabilities Monitoring Network"
"addm-Autism & Developmental Disabilities Monitoring Network" ...
 $ Asian.or.Pacific.Islander.Prevalence: chr  "No data" "No data" "No data"
"No data" ...
 $ Asian.or.Pacific.Islander.Lower.CI  : chr  "No data" "No data" "No data"
"No data" ...
 $ Asian.or.Pacific.Islander.Upper.CI  : chr  "No data" "No data" "No data"
"No data" ...
```

Quiz:

Obtain feature/column names and column index of dataframe: ASD_State

In [15]:
```
# Write your code below and press Shift+Enter to execute
```

Double-click **here** for the solution.

R Fundamentals - Work with dataframe

Access column 1 as a named list:

In [16]:
```
# use column index:
ASD_National[1]
```

Source
addm
addm
addm
addm
addm
addm
addm
addm
nsch
nsch
nsch
nsch

In [17]:
```
typeof(ASD_National[1])
```

'list'

In [18]:
```
ASD_National[1]$Source
```

'addm' 'addm' 'addm' 'addm' 'addm' 'addm' 'addm' 'addm' 'nsch' 'nsch' 'nsch' 'nsch'
'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped'
'sped' 'sped' 'sped' 'sped' 'medi' 'medi' 'medi' 'medi' 'medi' 'medi' 'medi' 'medi' 'medi'
'medi' 'medi' 'medi' 'medi'

In [19]:
```
typeof(ASD_National[1]$Source)
```

'character'

In [20]:
```
# use column name:
ASD_National["Source"]
```

Source
addm
addm
addm
addm
addm
addm
addm
addm
nsch
nsch
nsch
nsch

In [21]:
```
ASD_National['Source']$Source
```

'addm' 'addm' 'addm' 'addm' 'addm' 'addm' 'addm' 'addm' 'nsch' 'nsch' 'nsch' 'nsch'
'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped'
'sped' 'sped' 'sped' 'sped' 'medi' 'medi' 'medi' 'medi' 'medi' 'medi' 'medi' 'medi' 'medi'
'medi' 'medi' 'medi' 'medi'

Access column 1 as a set of string/chr:

In [22]:
```
ASD_National[, 1]
```

'addm' 'addm' 'addm' 'addm' 'addm' 'addm' 'addm' 'addm' 'nsch' 'nsch' 'nsch' 'nsch'
'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped'
'sped' 'sped' 'sped' 'sped' 'medi' 'medi' 'medi' 'medi' 'medi' 'medi' 'medi' 'medi' 'medi'
'medi' 'medi' 'medi' 'medi'

In [23]:
```
# or
ASD_National[, "Source"]
```

'addm' 'addm' 'addm' 'addm' 'addm' 'addm' 'addm' 'addm' 'nsch' 'nsch' 'nsch' 'nsch'
'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped'
'sped' 'sped' 'sped' 'sped' 'medi' 'medi' 'medi' 'medi' 'medi' 'medi' 'medi' 'medi' 'medi'
'medi' 'medi' 'medi' 'medi'

In [24]:
```
# or
ASD_National$Source
```

'addm' 'addm' 'addm' 'addm' 'addm' 'addm' 'addm' 'addm' 'nsch' 'nsch' 'nsch' 'nsch'
'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped'
'sped' 'sped' 'sped' 'sped' 'medi' 'medi' 'medi' 'medi' 'medi' 'medi' 'medi' 'medi' 'medi'
'medi' 'medi' 'medi' 'medi'

In [25]:
```
typeof(ASD_National$Source)
```

'character'

Count number of elements in a object:

In [26]:
```
length(ASD_National) # number of features/columns
```

26

In [27]:
```
length(ASD_National[1, ]) # number of elements(columns) in row 1
```

26

In [28]:
```
length(ASD_National[, 1]) # number of elements(rows) in column 1
```

42

In [29]:
```
length(ASD_National[, "Source"]) # same as above
```

42

In [30]:
```
length(ASD_National$Source) # number of elements in chr list
```

42

Access elements from dataframe

In [31]:
```
# using column index
ASD_National[1][1, ]
```

'addm'

In [32]:
```
ASD_National[1][11, ]
```

'nsch'

In [33]:
```
ASD_National[1][11:20, ]
```

'nsch' 'nsch' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped'

In [34]:
```
# using column name
ASD_National["Source"][1, ]
```

'addm'

In [35]:
```
ASD_National["Source"][11, ]
```

'nsch'

In [36]:
```
ASD_National["Source"][11:20, ]
```

'nsch' 'nsch' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped'

Access elements from dataframe

In [37]:
```
# using column index
ASD_National[, 1][1]
```

'addm'

In [38]:
```
ASD_National[, 1][11]
```

'nsch'

In [39]:
```
ASD_National[, 1][11:20]
```

'nsch' 'nsch' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped'

In [40]:
```
# using column name
ASD_National[, "Source"][1]
```

'addm'

In [41]:
```
# using column name
ASD_National[, "Source"][11]
```

'nsch'

In [42]:
```
# using column name
ASD_National[, "Source"][11:20]
```

'nsch' 'nsch' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped'

In [43]:
```
# using $ operator
ASD_National$Source[1]
```

'addm'

In [44]:
```
ASD_National$Source[11]
```

'nsch'

In [45]:
```
ASD_National$Source[11:20]
```

'nsch' 'nsch' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped' 'sped'

Access elements of different column:

In [46]:
```
cbind(names(ASD_National), c(1:length(names(ASD_National))))
```

Source	1
Year	2
Prevalence	3
Upper.CI	4
Lower.CI	5
Prevalence_dup	6
Source_Full1	7
Source_Full2	8
Male.Prevalence	9
Male.Lower.CI	10
Male.Upper.CI	11
Female.Prevalence	12
Female.Lower.CI	13

In [47]:
```
ASD_National[1, 1] # row 1, column 1: "Source"
```

'addm'

In [48]:
```
ASD_National[10, 1] # row 10, column 1: "Source"
```

'nsch'

In [49]:
```
ASD_National[1, 3] # row 1, column 3: "Prevalence"
```

6.7

In [50]:
```
ASD_National[10, 3] # row 10, column 3: "Prevalence"
```

16.2

In [51]:
```
ASD_National[1:10, 1:3] # row 1 to 10 from column 1 to 3
```

Source	Year	Prevalence
addm	2000	6.7
addm	2002	6.6
addm	2004	8.0
addm	2006	9.0
addm	2008	11.3
addm	2010	14.7
addm	2012	14.8
addm	2014	16.8
nsch	2004	9.5
nsch	2008	16.2

In [52]:
```
# or using columns names
ASD_National[1:10, c('Source', 'Year', 'Prevalence')]
```

Source	Year	Prevalence
addm	2000	6.7
addm	2002	6.6
addm	2004	8.0
addm	2006	9.0
addm	2008	11.3
addm	2010	14.7
addm	2012	14.8
addm	2014	16.8
nsch	2004	9.5
nsch	2008	16.2

In [53]:
```
ASD_National[c(1:10, 20, 30:35), c(1:3, 9, 12)] # row 1 to 10, 20, and 20 to 2
```

	Source	Year	Prevalence	Male.Prevalence	Female.Prevalence
1	addm	2000	6.7	No data	No data
2	addm	2002	6.6	11.5	2.7
3	addm	2004	8.0	12.9	2.9
4	addm	2006	9.0	14.5	3.2
5	addm	2008	11.3	18.4	4
6	addm	2010	14.7	23.7	5.3
7	addm	2012	14.8	23.4	5.2
8	addm	2014	16.8	26.6	6.6
9	nsch	2004	9.5		
10	nsch	2008	16.2		
20	sped	2007	5.4		
30	medi	2000	2.3		
31	medi	2001	2.6		
32	medi	2002	2.8		
33	medi	2003	3.0		
34	medi	2004	3.5		
35	medi	2005	3.9		

[Tips] We notice missing data from above.

R Fundamentals - Process missing data

Count missing values in dataframe:

In [54]:
```
sum(is.na(ASD_National)) # No missing data recognised by R (NA)
```

0

In [55]:
```
sum(is.na(ASD_State)) # Some missing data recognised by R (NA)
```

14454

Empty string, "No data" are not considered as missing value by R, thus we need to handle them manually.

In [56]:
```
# Define several offending strings
na_strings <- c("", "No data", "NA", "N A", "N / A", "N/A", "N/ A", "Not Avail
```

In [57]:
```
# Load required function from packages:
if(!require(naniar)){install.packages("naniar")}
library(naniar)
if(!require(dplyr)){install.packages("dplyr")}
library(dplyr)
```

```
Loading required package: naniar
Registered S3 methods overwritten by 'ggplot2':
  method         from
  [.quosures     rlang
  c.quosures     rlang
  print.quosures rlang
Loading required package: dplyr

Attaching package: ‘dplyr’

The following objects are masked from ‘package:stats’:

    filter, lag

The following objects are masked from ‘package:base’:

    intersect, setdiff, setequal, union
```

In [58]:
```
# Uncomment below to show help
# ?replace_with_na_all # Documentation
```

Replace these defined missing/offending values to R's internal NA

In [59]:
```
# "~.x" is a reserved keyword of this function:
ASD_National = replace_with_na_all(ASD_National, condition = ~.x %in% na_strin
```

In [60]:
```
# Count missing values (R's internal NA) in dataframe:
sum(is.na(ASD_National))
```

650

R Fundamentals - Process invalid characters

Remove invalid unicode char/string: \x92

In [61]:
```
ASD_National$Source_Full1[ASD_National$Source_Full1 == "National Survey of Chi
"National Survey of Children's Health"
```

In [62]:
```
ASD_National$Source_Full2[ASD_National$Source_Full2 == "nsch-National Survey o
"nsch-National Survey of Children's Health"
```

R Fundamentals - Delete/Drop dataframe variable

Delete/Drop duplicate variable: Prevalence_dup

In [63]:
```
drop <- c("Prevalence_dup", "Dummy Variable Name")
```

In [64]:
```
ASD_National = ASD_National[, !(names(ASD_National) %in% drop)] # Recall Dataf
```

R Fundamentals - Create/Add dataframe variable

Create one new variable: Source_UC by converting to uppercase letters

In [65]:
```
ASD_National$Source_UC <- paste(toupper(ASD_National$Source))
```

Create one new variable: Source_Full3 by combining Source and Source_Full1

In [66]:
```
ASD_National$Source_Full3 <- paste(toupper(ASD_National$Source), ASD_National$
```

Create one new ordinal categorical variable: Prevalence_Rank2 ("Low", "High") by binning Prevalence

In [67]:
```
# Recode Risk into category from Prevalence

# Low [0, 5)
# High [5, +oo)

ASD_National$Prevalence_Risk2[ASD_National$Prevalence < 5] = "Low"
ASD_National$Prevalence_Risk2[ASD_National$Prevalence >= 5 ] = "High"
#
head(ASD_National)
```

```
Warning message:
“Unknown or uninitialised column: 'Prevalence_Risk2'.”
```

Source	Year	Prevalence	Upper.CI	Lower.CI	Source_Full1	Source_Full2	Male.Prevalence	Male.Lower
addm	2000	6.7	7.0	6.3	Autism & Developmental Disabilities Monitoring Network	addm-Autism & Developmental Disabilities Monitoring Network	NA	N
addm	2002	6.6	6.8	6.3	Autism & Developmental Disabilities Monitoring Network	addm-Autism & Developmental Disabilities Monitoring Network	11.5	N
addm	2004	8.0	8.4	7.6	Autism & Developmental Disabilities Monitoring Network	addm-Autism & Developmental Disabilities Monitoring Network	12.9	1
addm	2006	9.0	9.3	8.6	Autism & Developmental Disabilities Monitoring Network	addm-Autism & Developmental Disabilities Monitoring Network	14.5	1
addm	2008	11.3	11.7	11.0	Autism & Developmental Disabilities Monitoring Network	addm-Autism & Developmental Disabilities Monitoring Network	18.4	1
addm	2010	14.7	15.1	14.3	Autism & Developmental Disabilities Monitoring Network	addm-Autism & Developmental Disabilities Monitoring Network	23.7	2

Create one new ordinal categorical variable: Prevalence_Rank4 ("Low", "Medium", "High", "Very High") by binning Prevalence

In [68]:
```
# Recode Risk into category from Prevalence

# Low [0, 5)
# Medium [5, 10)
# High [10, 20)
# Very High [20, +oo)

ASD_National$Prevalence_Risk4 = "Very High"
ASD_National$Prevalence_Risk4[ASD_National$Prevalence < 20 ] = "High"
ASD_National$Prevalence_Risk4[ASD_National$Prevalence < 10 ] = "Medium"
ASD_National$Prevalence_Risk4[ASD_National$Prevalence < 5] = "Low"
#
head(ASD_National)
```

Source	Year	Prevalence	Upper.CI	Lower.CI	Source_Full1	Source_Full2	Male.Prevalence	Male.Lower
addm	2000	6.7	7.0	6.3	Autism & Developmental Disabilities Monitoring Network	addm-Autism & Developmental Disabilities Monitoring Network	NA	N
addm	2002	6.6	6.8	6.3	Autism & Developmental Disabilities Monitoring Network	addm-Autism & Developmental Disabilities Monitoring Network	11.5	N
addm	2004	8.0	8.4	7.6	Autism & Developmental Disabilities Monitoring Network	addm-Autism & Developmental Disabilities Monitoring Network	12.9	1
addm	2006	9.0	9.3	8.6	Autism & Developmental Disabilities Monitoring Network	addm-Autism & Developmental Disabilities Monitoring Network	14.5	1
addm	2008	11.3	11.7	11.0	Autism & Developmental Disabilities Monitoring Network	addm-Autism & Developmental Disabilities Monitoring Network	18.4	1
addm	2010	14.7	15.1	14.3	Autism & Developmental Disabilities Monitoring Network	addm-Autism & Developmental Disabilities Monitoring Network	23.7	2

R Fundamentals - Convert to correct data types

Review data structure and variable names:

In [69]:
```
str(ASD_National)
cbind(names(ASD_National), c(1:length(names(ASD_National))))
```

```
Classes ‘tbl_df’, ‘tbl’ and 'data.frame':       42 obs. of  29 variables:
 $ Source                                : chr  "addm" "addm" "addm" "addm" ...
 $ Year                                  : int  2000 2002 2004 2006 2008 2010 2
012 2014 2004 2008 ...
 $ Prevalence                            : num  6.7 6.6 8 9 11.3 14.7 14.8 16.8
9.5 16.2 ...
 $ Upper.CI                              : num  7 6.8 8.4 9.3 11.7 15.1 15.2 1
7.3 12 18.1 ...
 $ Lower.CI                              : num  6.3 6.3 7.6 8.6 11 14.3 14.4 1
6.4 7.4 14.5 ...
 $ Source_Full1                          : chr  "Autism & Developmental Disabil
ities Monitoring Network" "Autism & Developmental Disabilities Monitoring Net
work" "Autism & Developmental Disabilities Monitoring Network" "Autism & Deve
lopmental Disabilities Monitoring Network" ...
 $ Source_Full2                          : chr  "addm-Autism & Developmental Di
sabilities Monitoring Network" "addm-Autism & Developmental Disabilities Moni
toring Network" "addm-Autism & Developmental Disabilities Monitoring Network"
"addm-Autism & Developmental Disabilities Monitoring Network" ...
 $ Male.Prevalence                       : chr  NA "11.5" "12.9" "14.5" ...
 $ Male.Lower.CI                         : chr  NA NA "12.2" "13.9" ...
 $ Male.Upper.CI                         : chr  NA NA "13.7" "15.1" ...
 $ Female.Prevalence                     : chr  NA "2.7" "2.9" "3.2" ...
 $ Female.Lower.CI                       : chr  NA NA "2.6" "2.9" ...
 $ Female.Upper.CI                       : chr  NA NA "3.3" "3.5" ...
 $ Non.hispanic.white.Prevalence         : chr  NA "7.7" "9.7" "9.9" ...
 $ Non.hispanic.white.Lower.CI           : chr  NA NA "9.1" "9.4" ...
 $ Non.hispanic.white.Upper.CI           : chr  NA NA "10.4" "10.4" ...
 $ Non.hispanic.black.Prevalence         : chr  NA "6.5" "6.9" "7.2" ...
 $ Non.hispanic.black.Lower.CI           : chr  NA NA "6.2" "6.6" ...
 $ Non.hispanic.black.Upper.CI           : chr  NA NA "7.6" "7.8" ...
 $ Hispanic.Prevalence                   : chr  NA NA "6.2" "5.9" ...
 $ Hispanic.Lower.CI                     : chr  NA NA "5" "5.3" ...
 $ Hispanic.Upper.CI                     : chr  NA NA "7.5" "6.6" ...
 $ Asian.or.Pacific.Islander.Prevalence: chr  NA NA NA NA ...
 $ Asian.or.Pacific.Islander.Lower.CI  : chr  NA NA NA NA ...
 $ Asian.or.Pacific.Islander.Upper.CI  : chr  NA NA NA NA ...
 $ Source_UC                             : chr  "ADDM" "ADDM" "ADDM" "ADDM" ...
 $ Source_Full3                          : chr  "ADDM Autism & Developmental Di
sabilities Monitoring Network" "ADDM Autism & Developmental Disabilities Moni
toring Network" "ADDM Autism & Developmental Disabilities Monitoring Network"
"ADDM Autism & Developmental Disabilities Monitoring Network" ...
 $ Prevalence_Risk2                      : chr  "High" "High" "High" "High" ...
 $ Prevalence_Risk4                      : chr  "Medium" "Medium" "Medium" "Med
ium" ...
```

Source	1
Year	2
Prevalence	3
Upper.CI	4
Lower.CI	5
Source_Full1	6
Source_Full2	7
Male.Prevalence	8
Male.Lower.CI	9
Male.Upper.CI	10
Female.Prevalence	11

Female.Lower.CI	12
Female.Upper.CI	13
Non.hispanic.white.Prevalence	14
Non.hispanic.white.Lower.CI	15
Non.hispanic.white.Upper.CI	16
Non.hispanic.black.Prevalence	17
Non.hispanic.black.Lower.CI	18
Non.hispanic.black.Upper.CI	19
Hispanic.Prevalence	20
Hispanic.Lower.CI	21
Hispanic.Upper.CI	22
Asian.or.Pacific.Islander.Prevalence	23
Asian.or.Pacific.Islander.Lower.CI	24
Asian.or.Pacific.Islander.Upper.CI	25
Source_UC	26
Source_Full3	27
Prevalence_Risk2	28
Prevalence_Risk4	29

Convert Prevalence and CIs from categorical/chr to numeric, column 8 to 25

In [70]:
```
ix <- 8:25 # define an index
# apply()
ASD_National[ix] <- apply(ASD_National[ix], 2, as.numeric) # "2" meand column-
```

In [71]:
```
# Uncomment below to show help
# ?apply # Documentation
```

In [72]:
```
# or lapply()
ASD_National[ix] <- lapply(ASD_National[ix], as.numeric) # column-wise
```

In [73]:
```
# Uncomment below to show help
# ?lapply # Documentation
```

Convert Source from categorical/chr to categorical/factor

In [74]:
```
ix <- c(1, 6, 7, 26, 27) # define an index
ASD_National[ix] <- lapply(ASD_National[ix], as.factor)
```

Create new ordered factor Year_Factor from Year

In [75]:
```
ASD_National$Year_Factor <- factor(ASD_National$Year, ordered = TRUE)
```

In [76]:
```
# Observe the difference of 'Levels' in below two factors
ASD_National$Year_Factor # Ordinal categorical variable
str(ASD_National$Year_Factor)

ASD_National$Source # Nominal categorical variable
str(ASD_National$Source)
```

2000 2002 2004 2006 2008 2010 2012 2014 2004 2008 2012 2016 2000 2001
2002 2003 2004 2005 2006 2007 2008 2009 2010 2011 2012 2013 2014 2015
2016 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010 2011 2012

▶ **Levels**:

```
 Ord.factor w/ 17 levels "2000"<"2001"<..: 1 3 5 7 9 11 13 15 5 9 ...
```

addm addm addm addm addm addm addm addm nsch nsch nsch nsch sped sped
sped sped sped sped sped sped sped sped sped sped sped sped sped sped
sped medi medi medi medi medi medi medi medi medi medi medi medi medi

▶ **Levels**:

```
 Factor w/ 4 levels "addm","medi",..: 1 1 1 1 1 1 1 1 3 3 ...
```

Convert Prevalence_Rank2 & Prevalence_Rank4 to ordered factor

In [77]:
```
# Convert to factor
ASD_National$Prevalence_Risk2 = factor(ASD_National$Prevalence_Risk2, ordered=
                                       levels=c("Low", "High"))
# Convert to factor
ASD_National$Prevalence_Risk4 = factor(ASD_National$Prevalence_Risk4, ordered=
                                       levels=c("Low", "Medium", "High", "
```

In [78]:
```
# Optionally, below is manual conversion examples:
# ASD_National$Male.Prevalence = as.numeric(ASD_National$Male.Prevalence)
# ASD_National$Source = as.factor(ASD_National$Source)
# ASD_National$Prevalence_Risk2 = factor(ASD_National$Prevalence_Risk2, ordere
# ASD_National$Prevalence_Risk4 = factor(ASD_National$Prevalence_Risk4, ordere
```

Optionally, export the processed dataframe data to CSV file.

In [79]:
```
write.csv(ASD_National, file = "../dataset/ADV_ASD_National_R.csv", row.names
```

In [80]:
```
# Read back in above saved file:
# ASD_National <- read.csv("../dataset/ADV_ASD_National_R.csv")
# ASD_National$Year_Factor <- factor(ASD_National$Year_Factor, ordered = TRUE)
```

Data Summarization

Data Summarization - High Level Data Summary

In [81]:
```
summary(ASD_National)
```

```
 Source        Year        Prevalence        Upper.CI         Lower.CI
addm: 8   Min.   :2000   Min.   : 1.800   Min.   : 1.800   Min.   : 1.700
medi:13   1st Qu.:2004   1st Qu.: 3.950   1st Qu.: 3.950   1st Qu.: 3.875
nsch: 4   Median :2008   Median : 6.650   Median : 6.900   Median : 6.350
sped:17   Mean   :2007   Mean   : 7.952   Mean   : 8.207   Mean   : 7.712
          3rd Qu.:2011   3rd Qu.: 9.725   3rd Qu.:10.350   3rd Qu.: 9.625
          Max.   :2016   Max.   :29.200   Max.   :30.700   Max.   :27.700

                                                 Source_Full1
Autism & Developmental Disabilities Monitoring Network: 8
Medicaid                                                :13
National Survey of Children's Health                    : 4
Special Education Child Count                           :17



                                                    Source_Full2
addm-Autism & Developmental Disabilities Monitoring Network: 8
medi-Medicaid                                                :13
nsch-National Survey of Children's Health                    : 4
sped-Special Education Child Count                           :17



Male.Prevalence Male.Lower.CI   Male.Upper.CI   Female.Prevalence
Min.   :11.50   Min.   :12.20   Min.   :13.70   Min.   :2.700
1st Qu.:13.70   1st Qu.:14.85   1st Qu.:16.07   1st Qu.:3.050
Median :18.40   Median :20.20   Median :21.55   Median :4.000
Mean   :18.71   Mean   :19.22   Mean   :20.62   Mean   :4.271
3rd Qu.:23.55   3rd Qu.:22.93   3rd Qu.:24.32   3rd Qu.:5.250
Max.   :26.60   Max.   :25.80   Max.   :27.40   Max.   :6.600
NA's   :35      NA's   :36      NA's   :36      NA's   :35
Female.Lower.CI Female.Upper.CI Non.hispanic.white.Prevalence
Min.   :2.600   Min.   :3.300   Min.   : 7.70
1st Qu.:3.100   1st Qu.:3.700   1st Qu.: 9.80
Median :4.300   Median :4.950   Median :12.00
Mean   :4.217   Mean   :4.900   Mean   :12.51
3rd Qu.:4.975   3rd Qu.:5.675   3rd Qu.:15.55
Max.   :6.200   Max.   :7.000   Max.   :17.20
NA's   :36      NA's   :36      NA's   :35
Non.hispanic.white.Lower.CI Non.hispanic.white.Upper.CI
Min.   : 9.100              Min.   :10.40
1st Qu.: 9.925              1st Qu.:10.93
Median :13.100              Median :14.20
Mean   :12.733              Mean   :13.88
3rd Qu.:15.075              3rd Qu.:16.20
Max.   :16.500              Max.   :17.80
NA's   :36                  NA's   :36
Non.hispanic.black.Prevalence Non.hispanic.black.Lower.CI
Min.   : 6.50                 Min.   : 6.200
1st Qu.: 7.05                 1st Qu.: 7.325
Median :10.20                 Median :10.500
Mean   :10.31                 Mean   :10.200
3rd Qu.:12.70                 3rd Qu.:12.100
Max.   :16.00                 Max.   :15.100
NA's   :35                    NA's   :36
Non.hispanic.black.Upper.CI Hispanic.Prevalence Hispanic.Lower.CI
Min.   : 7.600              Min.   : 5.900      Min.   : 5.000
1st Qu.: 8.575              1st Qu.: 6.625      1st Qu.: 5.775
Median :12.000              Median : 9.000      Median : 8.300
Mean   :11.700              Mean   : 9.150      Mean   : 8.333
3rd Qu.:13.700              3rd Qu.:10.625      3rd Qu.: 9.850
Max.   :16.900              Max.   :14.000      Max.   :13.100
```

```
NA's   :36                  NA's   :36          NA's   :36
Hispanic.Upper.CI Asian.or.Pacific.Islander.Prevalence
Min.   : 6.600    Min.   : 9.70
1st Qu.: 7.775    1st Qu.:10.97
Median : 9.750    Median :11.85
Mean   :10.017    Mean   :11.72
3rd Qu.:11.425    3rd Qu.:12.60
Max.   :14.900    Max.   :13.50
NA's   :36        NA's   :38
Asian.or.Pacific.Islander.Lower.CI Asian.or.Pacific.Islander.Upper.CI
Min.   : 8.10                      Min.   :11.60
1st Qu.: 9.45                      1st Qu.:12.72
Median :10.30                      Median :13.65
Mean   :10.12                      Mean   :13.57
3rd Qu.:10.97                      3rd Qu.:14.50
Max.   :11.80                      Max.   :15.40
NA's   :38                         NA's   :38
Source_UC                                                          Source_Full3
ADDM: 8   ADDM Autism & Developmental Disabilities Monitoring Network: 8
MEDI:13   MEDI Medicaid                                                    :13
NSCH: 4   NSCH National Survey of Children's Health                       : 4
SPED:17   SPED Special Education Child Count                             :17


Prevalence_Risk2  Prevalence_Risk4  Year_Factor
Low :14           Low      :14      2004   : 4
High:28           Medium   :18      2008   : 4
                  High     : 8      2012   : 4
                  Very High: 2      2000   : 3
                                    2002   : 3
                                    2006   : 3
                                    (Other):21
```

Data Summarization - Summary of numeric variables

In [82]:
```
# Filter only numeric variables/columns
select_if(ASD_National, is.numeric) # library(dplyr)
```

Year	Prevalence	Upper.CI	Lower.CI	Male.Prevalence	Male.Lower.CI	Male.Upper.CI	Female.Prevalence
2000	6.7	7.0	6.3	NA	NA	NA	NA
2002	6.6	6.8	6.3	11.5	NA	NA	2.7
2004	8.0	8.4	7.6	12.9	12.2	13.7	2.9
2006	9.0	9.3	8.6	14.5	13.9	15.1	3.2
2008	11.3	11.7	11.0	18.4	17.7	19.0	4.0
2010	14.7	15.1	14.3	23.7	23.0	24.4	5.3
2012	14.8	15.2	14.4	23.4	22.7	24.1	5.2
2014	16.8	17.3	16.4	26.6	25.8	27.4	6.6
2004	9.5	12.0	7.4	NA	NA	NA	NA
2008	16.2	18.1	14.5	NA	NA	NA	NA
2012	21.2	22.3	20.1	NA	NA	NA	NA
2016	29.2	30.7	27.7	NA	NA	NA	NA
2000	1.8	1.8	1.7	NA	NA	NA	NA
2001	2.1	2.1	2.1	NA	NA	NA	NA
2002	2.6	2.6	2.6	NA	NA	NA	NA
2003	3.0	3.0	3.0	NA	NA	NA	NA
2004	3.6	3.6	3.5	NA	NA	NA	NA
2005	4.1	4.1	4.1	NA	NA	NA	NA
2006	4.8	4.8	4.7	NA	NA	NA	NA
2007	5.4	5.5	5.4	NA	NA	NA	NA
2008	6.2	6.2	6.2	NA	NA	NA	NA
2009	7.0	7.0	7.0	NA	NA	NA	NA
2010	7.7	7.7	7.7	NA	NA	NA	NA
2011	8.4	8.5	8.4	NA	NA	NA	NA
2012	9.1	9.2	9.1	NA	NA	NA	NA
2013	9.8	9.9	9.8	NA	NA	NA	NA
2014	10.5	10.5	10.5	NA	NA	NA	NA
2015	11.2	11.2	11.2	NA	NA	NA	NA
2016	11.9	11.9	11.9	NA	NA	NA	NA
2000	2.3	2.4	2.3	NA	NA	NA	NA
2001	2.6	2.6	2.6	NA	NA	NA	NA
2002	2.8	2.8	2.7	NA	NA	NA	NA
2003	3.0	3.0	3.0	NA	NA	NA	NA
2004	3.5	3.6	3.5	NA	NA	NA	NA
2005	3.9	3.9	3.8	NA	NA	NA	NA
2006	4.4	4.5	4.4	NA	NA	NA	NA
2007	5.1	5.1	5.0	NA	NA	NA	NA
2008	5.6	5.6	5.5	NA	NA	NA	NA
2009	5.9	5.9	5.9	NA	NA	NA	NA

Year	Prevalence	Upper.CI	Lower.CI	Male.Prevalence	Male.Lower.CI	Male.Upper.CI	Female.Prevalence
2010	6.4	6.4	6.4	NA	NA	NA	NA
2011	7.1	7.1	7.1	NA	NA	NA	NA
2012	8.2	8.3	8.2	NA	NA	NA	NA

In [83]:
```
# Data summarization
summary(select_if(ASD_National, is.numeric))
```

```
      Year        Prevalence        Upper.CI         Lower.CI
 Min.   :2000   Min.   : 1.800   Min.   : 1.800   Min.   : 1.700
 1st Qu.:2004   1st Qu.: 3.950   1st Qu.: 3.950   1st Qu.: 3.875
 Median :2008   Median : 6.650   Median : 6.900   Median : 6.350
 Mean   :2007   Mean   : 7.952   Mean   : 8.207   Mean   : 7.712
 3rd Qu.:2011   3rd Qu.: 9.725   3rd Qu.:10.350   3rd Qu.: 9.625
 Max.   :2016   Max.   :29.200   Max.   :30.700   Max.   :27.700

 Male.Prevalence Male.Lower.CI   Male.Upper.CI   Female.Prevalence
 Min.   :11.50   Min.   :12.20   Min.   :13.70   Min.   :2.700
 1st Qu.:13.70   1st Qu.:14.85   1st Qu.:16.07   1st Qu.:3.050
 Median :18.40   Median :20.20   Median :21.55   Median :4.000
 Mean   :18.71   Mean   :19.22   Mean   :20.62   Mean   :4.271
 3rd Qu.:23.55   3rd Qu.:22.93   3rd Qu.:24.32   3rd Qu.:5.250
 Max.   :26.60   Max.   :25.80   Max.   :27.40   Max.   :6.600
 NA's   :35      NA's   :36      NA's   :36      NA's   :35
 Female.Lower.CI Female.Upper.CI Non.hispanic.white.Prevalence
 Min.   :2.600   Min.   :3.300   Min.   : 7.70
 1st Qu.:3.100   1st Qu.:3.700   1st Qu.: 9.80
 Median :4.300   Median :4.950   Median :12.00
 Mean   :4.217   Mean   :4.900   Mean   :12.51
 3rd Qu.:4.975   3rd Qu.:5.675   3rd Qu.:15.55
 Max.   :6.200   Max.   :7.000   Max.   :17.20
 NA's   :36      NA's   :36      NA's   :35
 Non.hispanic.white.Lower.CI Non.hispanic.white.Upper.CI
 Min.   : 9.100              Min.   :10.40
 1st Qu.: 9.925              1st Qu.:10.93
 Median :13.100              Median :14.20
 Mean   :12.733              Mean   :13.88
 3rd Qu.:15.075              3rd Qu.:16.20
 Max.   :16.500              Max.   :17.80
 NA's   :36                  NA's   :36
 Non.hispanic.black.Prevalence Non.hispanic.black.Lower.CI
 Min.   : 6.50                 Min.   : 6.200
 1st Qu.: 7.05                 1st Qu.: 7.325
 Median :10.20                 Median :10.500
 Mean   :10.31                 Mean   :10.200
 3rd Qu.:12.70                 3rd Qu.:12.100
 Max.   :16.00                 Max.   :15.100
 NA's   :35                    NA's   :36
 Non.hispanic.black.Upper.CI Hispanic.Prevalence Hispanic.Lower.CI
 Min.   : 7.600              Min.   : 5.900      Min.   : 5.000
 1st Qu.: 8.575              1st Qu.: 6.625      1st Qu.: 5.775
 Median :12.000              Median : 9.000      Median : 8.300
 Mean   :11.700              Mean   : 9.150      Mean   : 8.333
 3rd Qu.:13.700              3rd Qu.:10.625      3rd Qu.: 9.850
 Max.   :16.900              Max.   :14.000      Max.   :13.100
 NA's   :36                  NA's   :36          NA's   :36
 Hispanic.Upper.CI Asian.or.Pacific.Islander.Prevalence
 Min.   : 6.600    Min.   : 9.70
 1st Qu.: 7.775    1st Qu.:10.97
 Median : 9.750    Median :11.85
 Mean   :10.017    Mean   :11.72
 3rd Qu.:11.425    3rd Qu.:12.60
 Max.   :14.900    Max.   :13.50
 NA's   :36        NA's   :38
 Asian.or.Pacific.Islander.Lower.CI Asian.or.Pacific.Islander.Upper.CI
 Min.   : 8.10                      Min.   :11.60
 1st Qu.: 9.45                      1st Qu.:12.72
 Median :10.30                      Median :13.65
 Mean   :10.12                      Mean   :13.57
 3rd Qu.:10.97                      3rd Qu.:14.50
```

```
Max.   :11.80                    Max.   :15.40
NA's   :38                       NA's   :38
```

[Tips] We notice missing data in a few Prevalence variables.

In [84]:
```
# Calculate average Prevalence, no error
mean(ASD_National$Prevalence)
mean(ASD_National$Prevalence[ASD_National$Source == 'addm'])
mean(ASD_National$Prevalence[ASD_National$Source == 'medi'])
mean(ASD_National$Prevalence[ASD_National$Source == 'nsch'])
mean(ASD_National$Prevalence[ASD_National$Source == 'sped'])
```

7.95238095238095

10.9875

4.67692307692308

19.025

6.42352941176471

In [85]:
```
# Calculate average Male.Prevalence, there is error!
mean(ASD_National$Male.Prevalence)
```

<NA>

In [86]:
```
# Because of NA, mean() cannot process, thus we use na.rm to ignore NAs
mean(ASD_National$Male.Prevalence, na.rm = TRUE)
```

18.7142857142857

In [87]:
```
mean(ASD_National$Female.Prevalence, na.rm = TRUE)
```

4.27142857142857

In [88]:
```
# Count occurrences of uniques values in a variable/column: number of rows (of
table(ASD_National$Year) # ?table
```

```
2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010 2011 2012 2013 2014 20
15
   3    2    3    2    4    2    3    2    4    2    3    2    4    1    2
1
2016
   2
```

Data Summarization - Summary of categorical variables

In [89]:
```
# List of categorical variables
names(select_if(ASD_National, is.factor)) # All categorical variables are fact
names(select_if(ASD_National, is.character)) # No categorical variable is char
```

'Source' 'Source_Full1' 'Source_Full2' 'Source_UC' 'Source_Full3' 'Prevalence_Risk2' 'Prevalence_Risk4' 'Year_Factor'

In [90]:
```
# Look at summary
summary(select_if(ASD_National, is.factor))
```

```
 Source                                                    Source_Full1
addm: 8   Autism & Developmental Disabilities Monitoring Network: 8
medi:13   Medicaid                                                :13
nsch: 4   National Survey of Children's Health                    : 4
sped:17   Special Education Child Count                           :17



                                                        Source_Full2 Source_UC
addm-Autism & Developmental Disabilities Monitoring Network: 8    ADDM: 8
medi-Medicaid                                                :13    MEDI:13
nsch-National Survey of Children's Health                    : 4    NSCH: 4
sped-Special Education Child Count                           :17    SPED:17



                                                        Source_Full3
ADDM Autism & Developmental Disabilities Monitoring Network: 8
MEDI Medicaid                                                :13
NSCH National Survey of Children's Health                    : 4
SPED Special Education Child Count                           :17



Prevalence_Risk2  Prevalence_Risk4  Year_Factor
Low :14           Low      :14      2004   : 4
High:28           Medium   :18      2008   : 4
                  High     : 8      2012   : 4
                  Very High: 2      2000   : 3
                                    2002   : 3
                                    2006   : 3
                                    (Other):21
```

In [91]:
```
summary(select_if(ASD_National, is.character))
```

```
< table of extent 0 x 0 >
```

In [92]:
```
# Count occurrences of uniques values in a variable/column
table(ASD_National$Source)
```

```
addm medi nsch sped
   8   13    4   17
```

In [93]:
```
table(ASD_National$Source_Full3)
```

```
ADDM Autism & Developmental Disabilities Monitoring Network
                                                           8
                                                MEDI Medicaid
                                                          13
                                NSCH National Survey of Children's Health
                                                           4
                                       SPED Special Education Child Count
                                                          17
```

In [94]:
```
table(ASD_National$Year_Factor)
```

```
2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010 2011 2012 2013 2014 20
15
   3    2    3    2    4    2    3    2    4    2    3    2    4    1    2
1
2016
   2
```

In [95]:
```
table(ASD_National$Prevalence) # numeric is also possible
```

```
 1.8  2.1  2.3  2.6  2.8    3  3.5  3.6  3.9  4.1  4.4  4.8  5.1  5.4  5.6
5.9
   1    1    1    2    1    2    1    1    1    1    1    1    1    1    1
1
 6.2  6.4  6.6  6.7    7  7.1  7.7    8  8.2  8.4    9  9.1  9.5  9.8 10.5 1
1.2
   1    1    1    1    1    1    1    1    1    1    1    1    1    1    1
1
11.3 11.9 14.7 14.8 16.2 16.8 21.2 29.2
   1    1    1    1    1    1    1    1
```

In [96]:
```
# Display unique values (levels) of a factor categorical
lapply(select_if(ASD_National, is.factor), levels)
```

$Source
'addm' 'medi' 'nsch' 'sped'

$Source_Full1
'Autism & Developmental Disabilities Monitoring Network' 'Medicaid'
'National Survey of Children\'s Health' 'Special Education Child Count'

$Source_Full2
'addm-Autism & Developmental Disabilities Monitoring Network' 'medi-Medicaid'
'nsch-National Survey of Children\'s Health' 'sped-Special Education Child Count'

$Source_UC
'ADDM' 'MEDI' 'NSCH' 'SPED'

$Source_Full3
'ADDM Autism & Developmental Disabilities Monitoring Network' 'MEDI Medicaid'
'NSCH National Survey of Children\'s Health' 'SPED Special Education Child Count'

$Prevalence_Risk2
'Low' 'High'

$Prevalence_Risk4
'Low' 'Medium' 'High' 'Very High'

$Year_Factor
'2000' '2001' '2002' '2003' '2004' '2005' '2006' '2007' '2008' '2009' '2010' '2011' '2012'
'2013' '2014' '2015' '2016'

In [97]:
```
# or using variable names
lapply(ASD_National[c('Source_UC', 'Year_Factor')], levels)
```

$Source_UC

'ADDM' 'MEDI' 'NSCH' 'SPED'

$Year_Factor

'2000' '2001' '2002' '2003' '2004' '2005' '2006' '2007' '2008' '2009' '2010' '2011' '2012' '2013' '2014' '2015' '2016'

In [98]:
```
# Pivot of counting occurrences
table(ASD_National$Source_Full3, ASD_National$Year) # table(ASD_National$Year,
```

	2000	2001	2002
ADDM Autism & Developmental Disabilities Monitoring Network	1	0	1
MEDI Medicaid	1	1	1
NSCH National Survey of Children's Health	0	0	0
SPED Special Education Child Count	1	1	1

	2003	2004	2005
ADDM Autism & Developmental Disabilities Monitoring Network	0	1	0
MEDI Medicaid	1	1	1
NSCH National Survey of Children's Health	0	1	0
SPED Special Education Child Count	1	1	1

	2006	2007	2008
ADDM Autism & Developmental Disabilities Monitoring Network	1	0	1
MEDI Medicaid	1	1	1
NSCH National Survey of Children's Health	0	0	1
SPED Special Education Child Count	1	1	1

	2009	2010	2011
ADDM Autism & Developmental Disabilities Monitoring Network	0	1	0
MEDI Medicaid	1	1	1
NSCH National Survey of Children's Health	0	0	0
SPED Special Education Child Count	1	1	1

	2012	2013	2014
ADDM Autism & Developmental Disabilities Monitoring Network	1	0	1
MEDI Medicaid	1	0	0
NSCH National Survey of Children's Health	1	0	0
SPED Special Education Child Count	1	1	1

	2015	2016
ADDM Autism & Developmental Disabilities Monitoring Network	0	0
MEDI Medicaid	0	0
NSCH National Survey of Children's Health	0	1
SPED Special Education Child Count	1	1

In [99]:
```
# Pivot of counting occurrences
table(ASD_National$Prevalence_Risk2, ASD_National$Source)

# Pivot of counting occurrences
table(ASD_National$Prevalence_Risk4, ASD_National$Source)
```

```
      addm medi nsch sped
  Low    0    7    0    7
  High   8    6    4   10
```

```
           addm medi nsch sped
  Low         0    7    0    7
  Medium      4    6    1    7
  High        4    0    1    3
  Very High   0    0    2    0
```

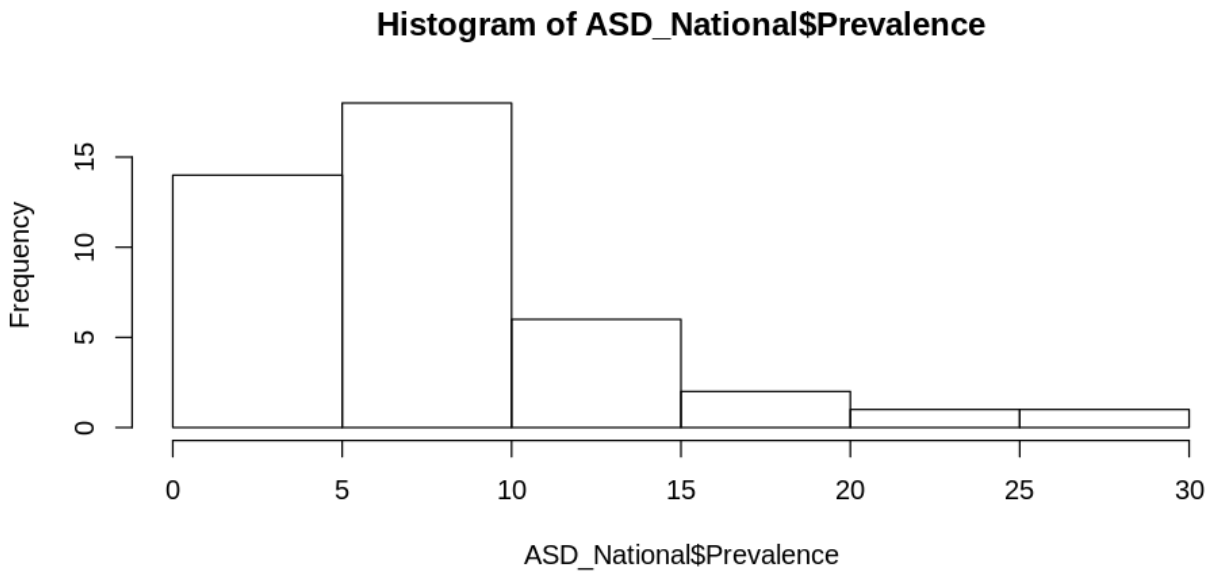
Data Visualisation (Base Graphic)

In [100]:
```
# library(repr)
# Adjust in-line plot size to M x N
options(repr.plot.width=8, repr.plot.height=4)
```

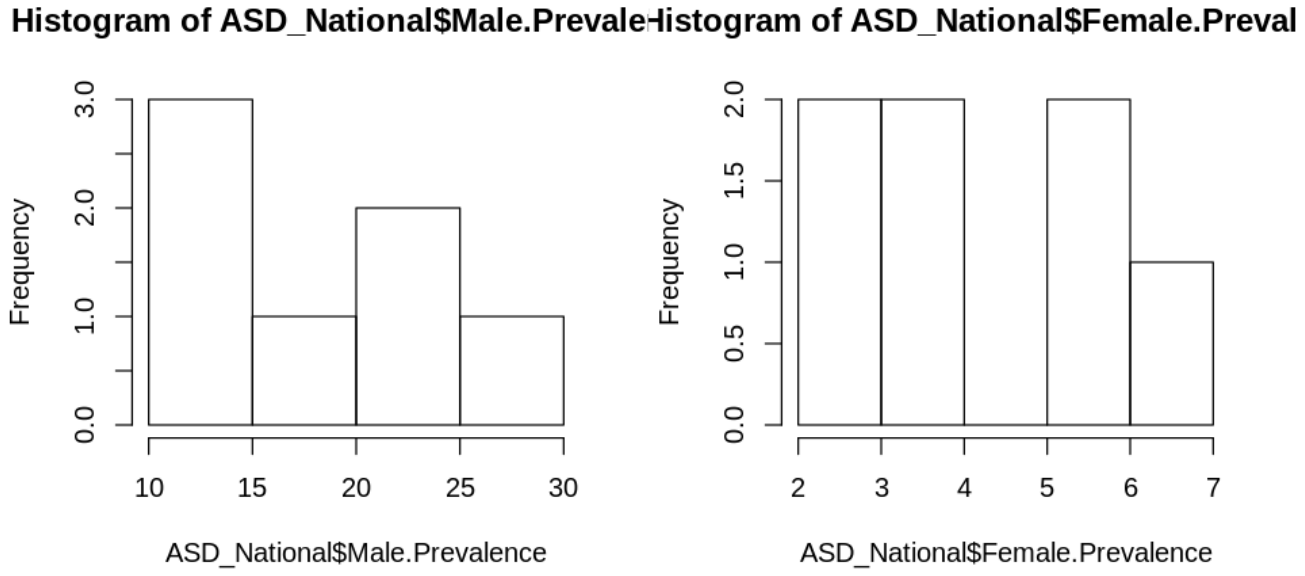
Data Visualisation (Base Graphic) - Histogram (distribution of binned continuous variable)

https://www.statmethods.net/graphs/density.html (https://www.statmethods.net/graphs/density.html)

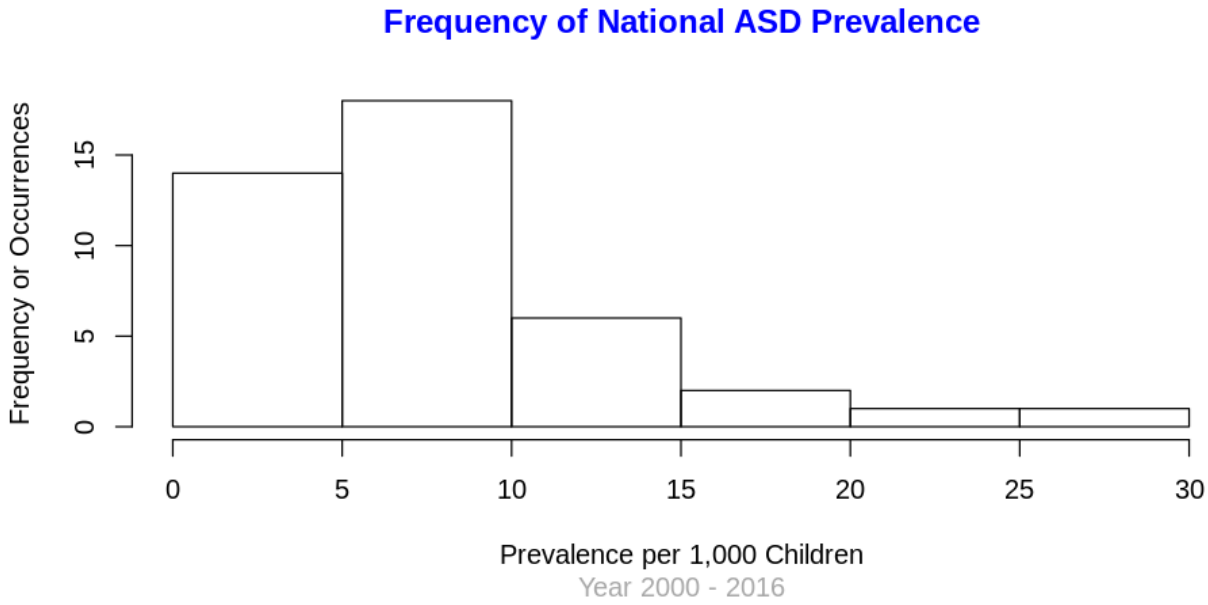
In [101]:
```
hist(ASD_National$Prevalence)
```



In [102]:
```
par(mfrow=c(1, 2)) # multiple plots on one page: row split to: 1,column split
hist(ASD_National$Male.Prevalence)
hist(ASD_National$Female.Prevalence)
par(mfrow=c(1, 1)) # Reset to one plot on one page
```



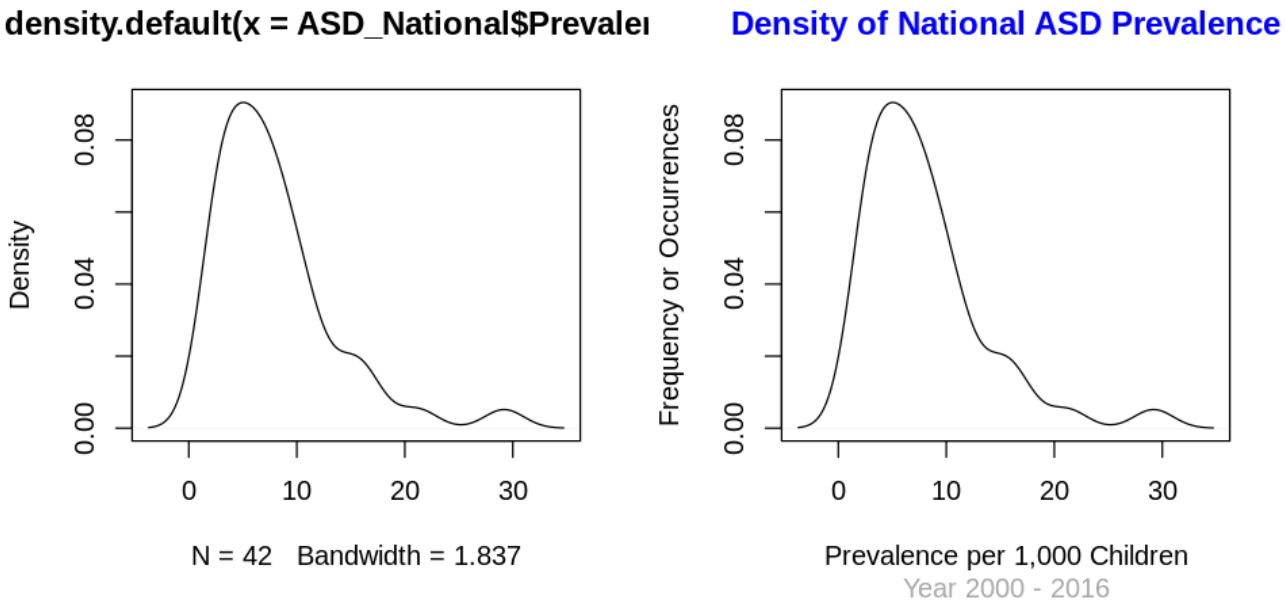
In [103]:
```
# Histogram with annotations
hist(ASD_National$Prevalence,
     main = "Frequency of National ASD Prevalence", # Chart title
     xlab = "Prevalence per 1,000 Children", # x axis label
     ylab = "Frequency or Occurrences",# y axis label
     sub  = "Year 2000 - 2016", # Chart subtitle at bottom
     col.main="blue", col.lab="black", col.sub="darkgrey") # Colours
```



Density plot (distribution for continuous variable normalized to 100% area under curve)

https://www.statmethods.net/graphs/density.html (https://www.statmethods.net/graphs/density.html)

In [104]:
```
par(mfrow=c(1, 2)) # multiple plots on one page: row split to: 1,column split

plot(density(ASD_National$Prevalence))

# Density plot with annotations
plot(density(ASD_National$Prevalence),
     main = "Density of National ASD Prevalence",
     xlab = "Prevalence per 1,000 Children",
     ylab = "Frequency or Occurrences",
     sub  = "Year 2000 - 2016",
     col.main="blue", col.lab="black", col.sub="darkgrey")

par(mfrow=c(1, 1))
```



Boxplot plot (median, 25% quantile,75% quantile)

https://www.statmethods.net/graphs/boxplot.html (https://www.statmethods.net/graphs/boxplot.html)

https://stats.stackexchange.com/questions/156778/percentile-vs-quantile-vs-quartile (https://stats.stackexchange.com/questions/156778/percentile-vs-quantile-vs-quartile)

0 quartile = 0 quantile = 0 percentile

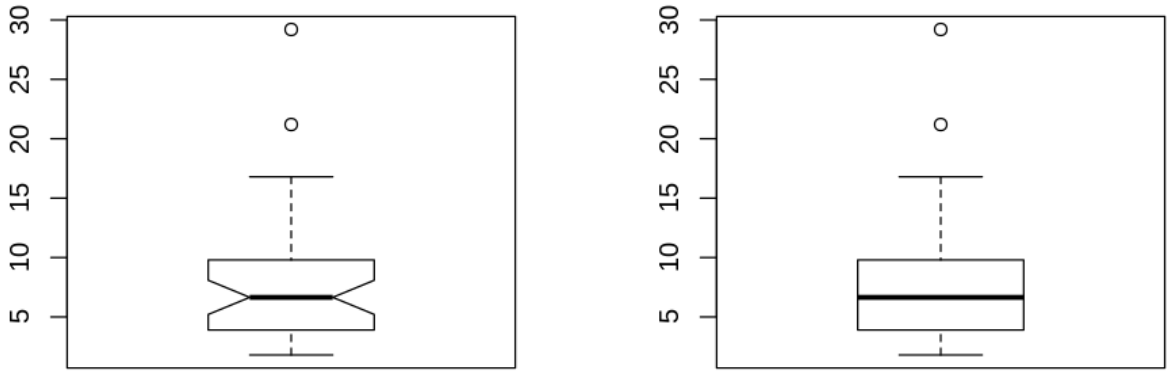
1 quartile = 0.25 quantile = 25 percentile

2 quartile = .5 quantile = 50 percentile (median)

3 quartile = .75 quantile = 75 percentile

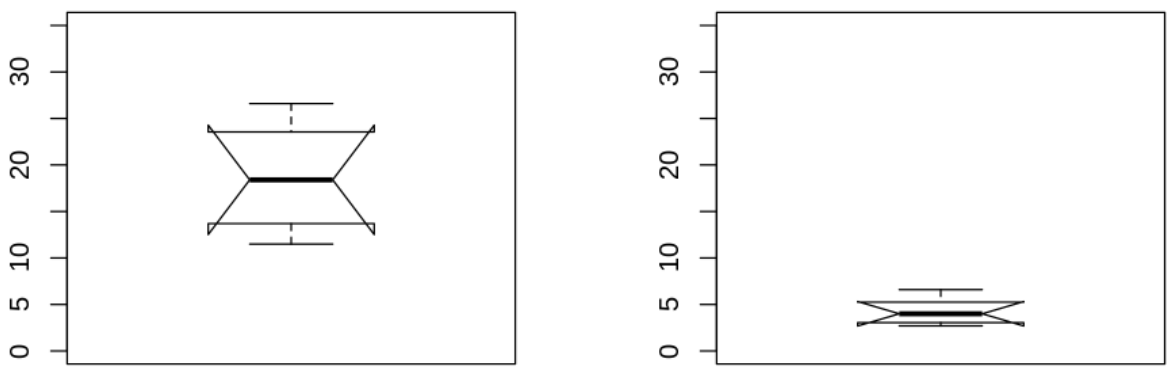
4 quartile = 1 quantile = 100 percentile

In [105]:
```
par(mfrow=c(1, 2)) # multiple plots on one page: row split to: 1,column split

# All children prevalence with and without 95% confidence side by side:
boxplot(ASD_National$Prevalence, notch = TRUE) # 95% confidence interval - a n
boxplot(ASD_National$Prevalence) # All children

par(mfrow=c(1, 1))
```



In [106]:
```
par(mfrow=c(1, 2)) # multiple plots on one page: row split to: 1,column split

# Male prevalence and Female prevalence side by side:
boxplot(ASD_National$Male.Prevalence, ylim = c(0, 35), notch = TRUE) # Male ch
boxplot(ASD_National$Female.Prevalence, ylim = c(0, 35), notch = TRUE) # Femal

par(mfrow=c(1, 1))
```

```
Warning message in bxp(list(stats = structure(c(11.5, 13.7, 18.4, 23.55, 26.
6), .Dim = c(5L, :
“some notches went outside hinges ('box'): maybe set notch=FALSE”Warning mess
age in bxp(list(stats = structure(c(2.7, 3.05, 4, 5.25, 6.6), .Dim = c(5L, :
“some notches went outside hinges ('box'): maybe set notch=FALSE”
```



In [107]:
```
# Display value ranges
# numeric:
range(ASD_National$Prevalence)
```

1.8 29.2

In [108]:
```
range(ASD_National$Year)
```

2000 2016

In [109]:
```
# categorical:
min(ASD_National$Year_Factor)
```

2000

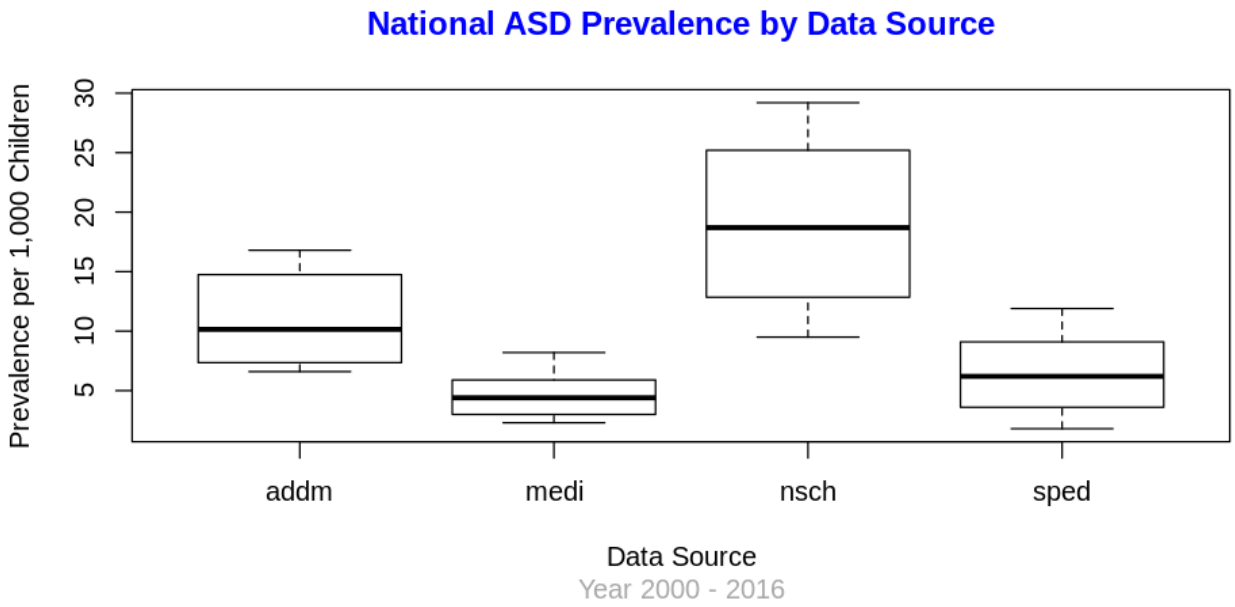
▶ **Levels**:

In [110]:
```
max(ASD_National$Year_Factor)
```

2016

▶ **Levels**:

In [111]:
```
# Create 'Prevalence' box plots break by 'Source'
boxplot(ASD_National$Prevalence ~ ASD_National$Source,
        main = "National ASD Prevalence by Data Source",
        xlab = "Data Source",
        ylab = "Prevalence per 1,000 Children",
        sub  = "Year 2000 - 2016",
        col.main="blue", col.lab="black", col.sub="darkgrey")
```



Quiz:

Set noth=TRUE to above boxplot. Are there overlapping among four data sources?

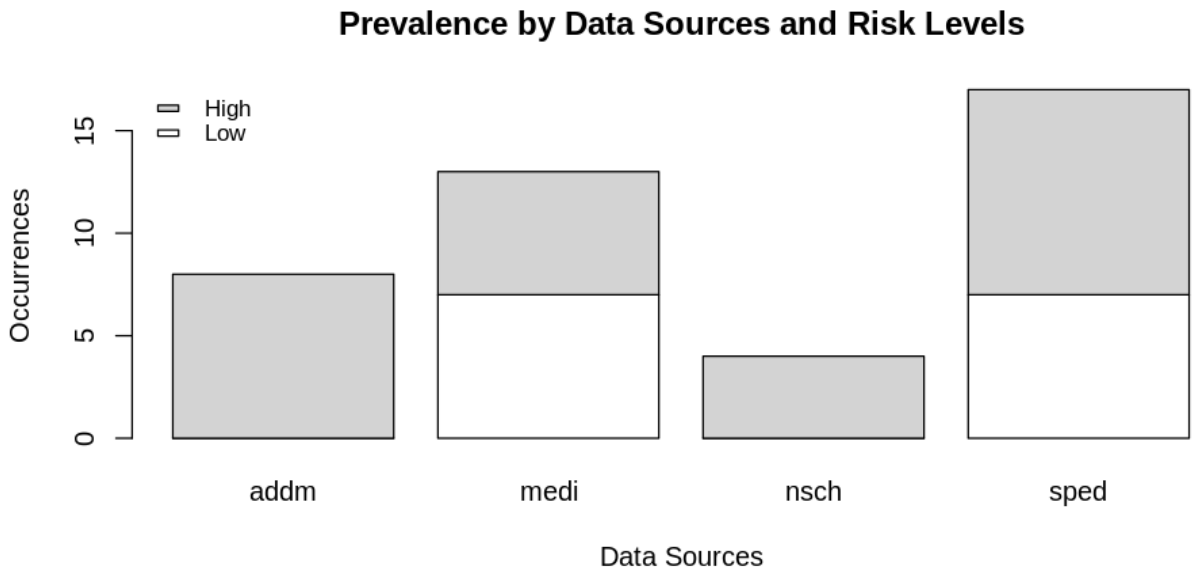
In [112]:
```
# Write your code below and press Shift+Enter to execute
```

Double-click **here** for the solution.

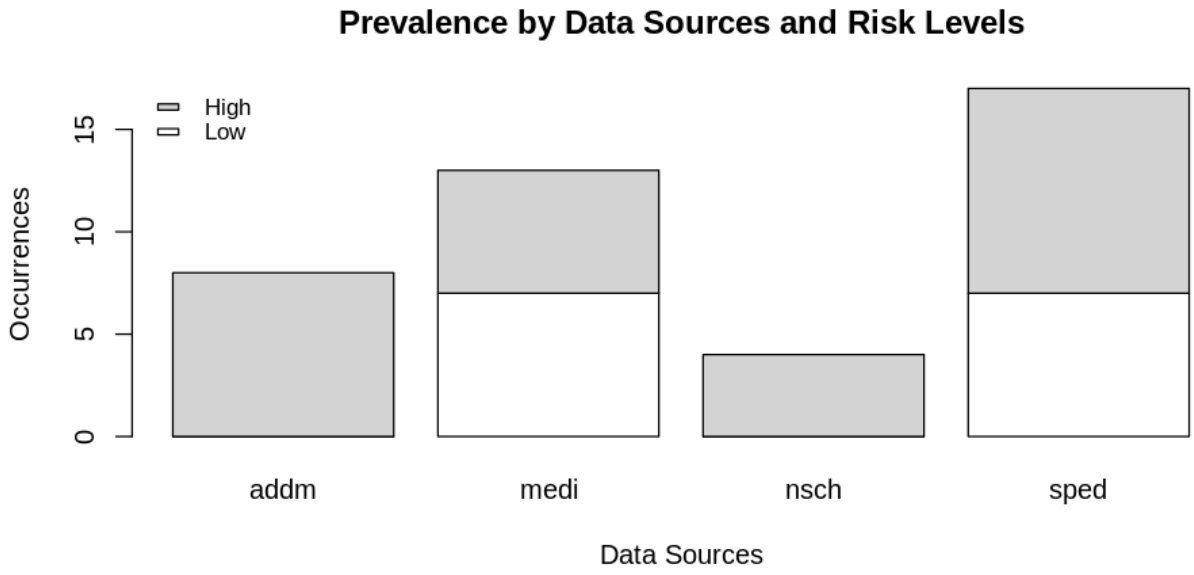
Data Visualisation (Base Graphic) - Bar plot

In [113]:
```
# Adjust in-line plot size to M x N
options(repr.plot.width=8, repr.plot.height=4)
```

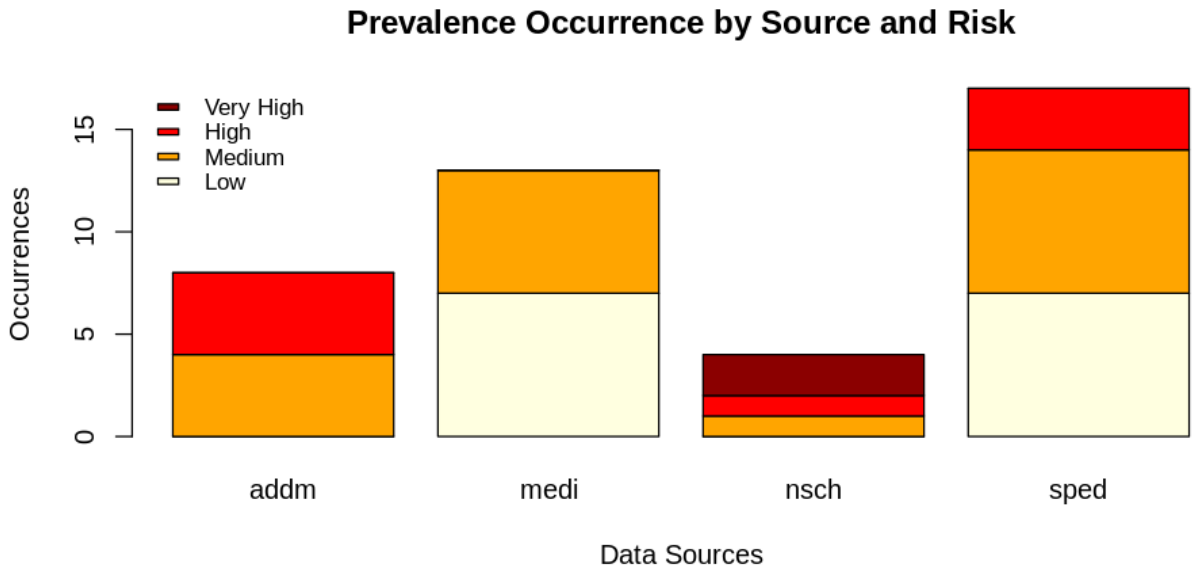
In [114]:
```
# ---------------------------------
# [National] Risk by Data Source
# ---------------------------------
# Create bar chart using R graphics
counts = table(ASD_National$Prevalence_Risk2, ASD_National$Source)
#counts = table(ASD_National$Source, ASD_National$Prevalence_Risk4)
barplot(counts,
        main="Prevalence by Data Sources and Risk Levels",
        xlab="Data Sources", col=c("white", "lightgrey"),
        ylab="Occurrences",
        legend = rownames(counts),
        args.legend = list(x="topleft", bty = "n", cex = 0.85, y.intersp=2))
```



In [115]:
```r
# ---------------------------------
# [National] Risk by Data Source
# ---------------------------------
# Create bar chart using R graphics
counts = table(ASD_National$Prevalence_Risk2, ASD_National$Source) # Count of
barplot(counts,
        main="Prevalence by Data Sources and Risk Levels",
        xlab="Data Sources",
        ylab="Occurrences",
        col=c("white", "lightgrey"),
        legend = rownames(counts),
        args.legend = list(x = "topleft", bty = "n", cex = 0.85, y.intersp = 2
```



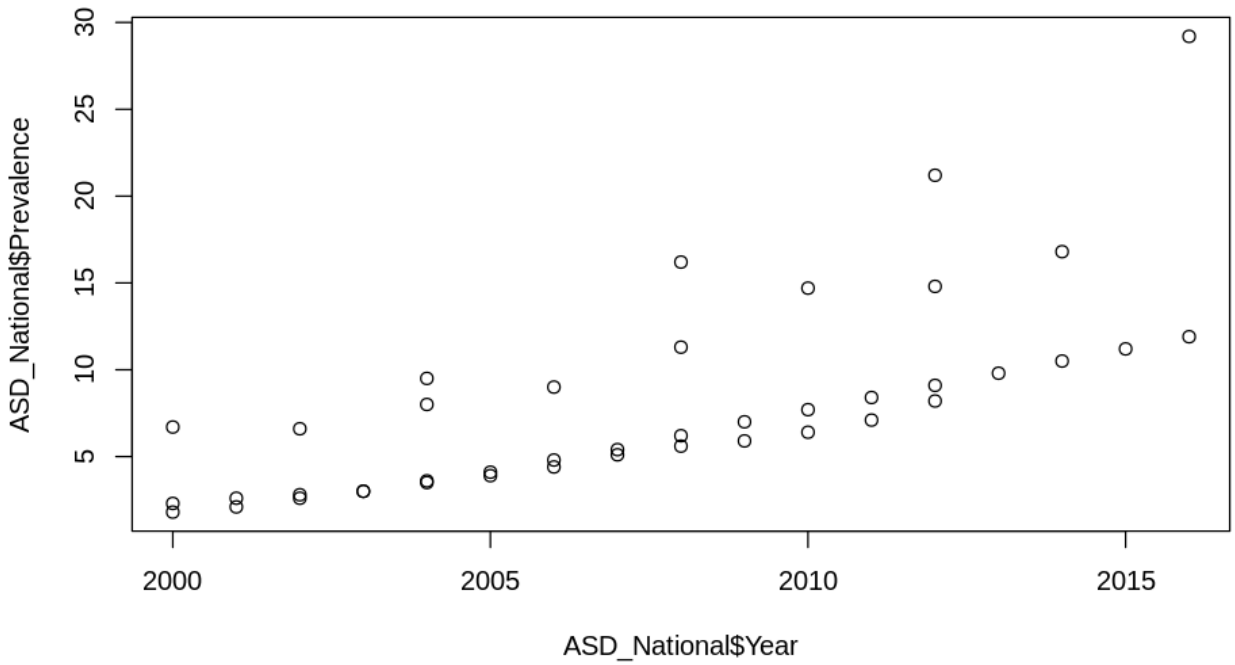
In [116]:
```
# ---------------------------------
# [National] Risk by Data Source
# ---------------------------------
# Create bar chart using R graphics
counts = table(ASD_National$Prevalence_Risk4, ASD_National$Source) # Count of
barplot(counts,
        main="Prevalence Occurrence by Source and Risk",
        xlab="Data Sources",
        ylab="Occurrences",
        col=c("lightyellow", "orange", "red","darkred"),
        legend = rownames(counts),
        args.legend = list(x = "topleft", bty = "n", cex = 0.85, y.intersp = 2
```



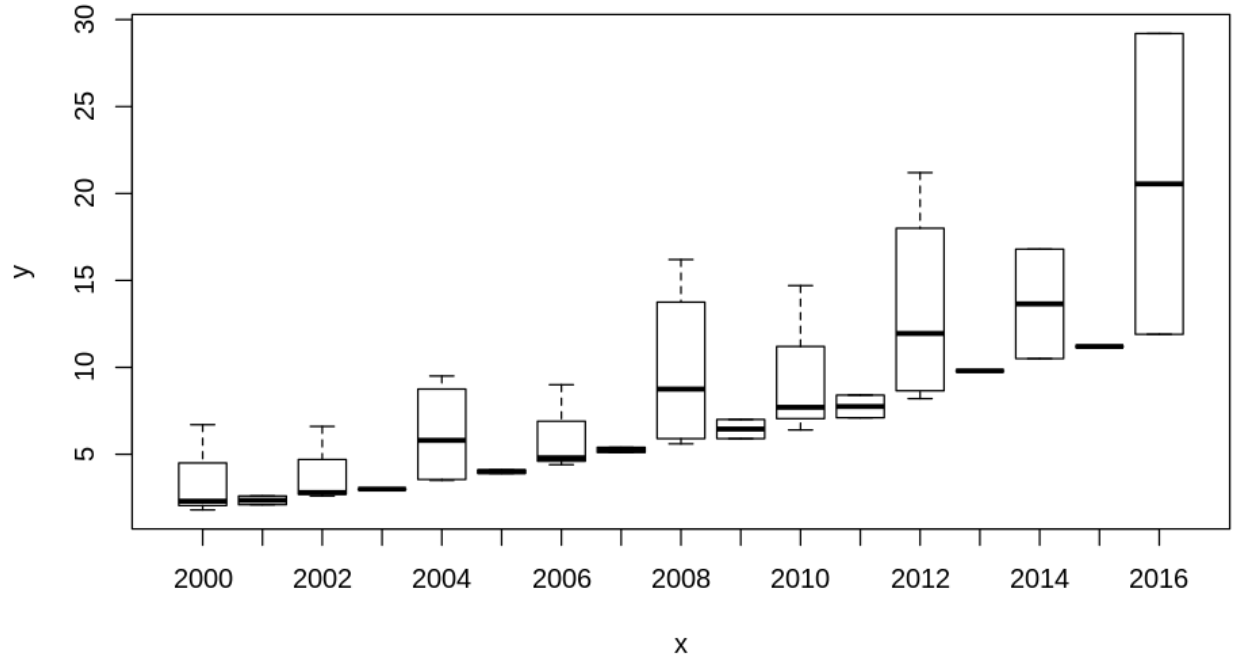
Data Visualisation (Base Graphic) - Line chart

In [117]:
```
# Adjust in-line plot size to M x N
options(repr.plot.width=8, repr.plot.height=5)
```

In [118]:
```
# ---------------------------------
# [National] < Prevalence has changed over Time >
# ---------------------------------
# Prevalence over Year
# Use Year        as x-axis: y value Prevalence is NOT aggregated for differen
plot(ASD_National$Year, ASD_National$Prevalence)
```



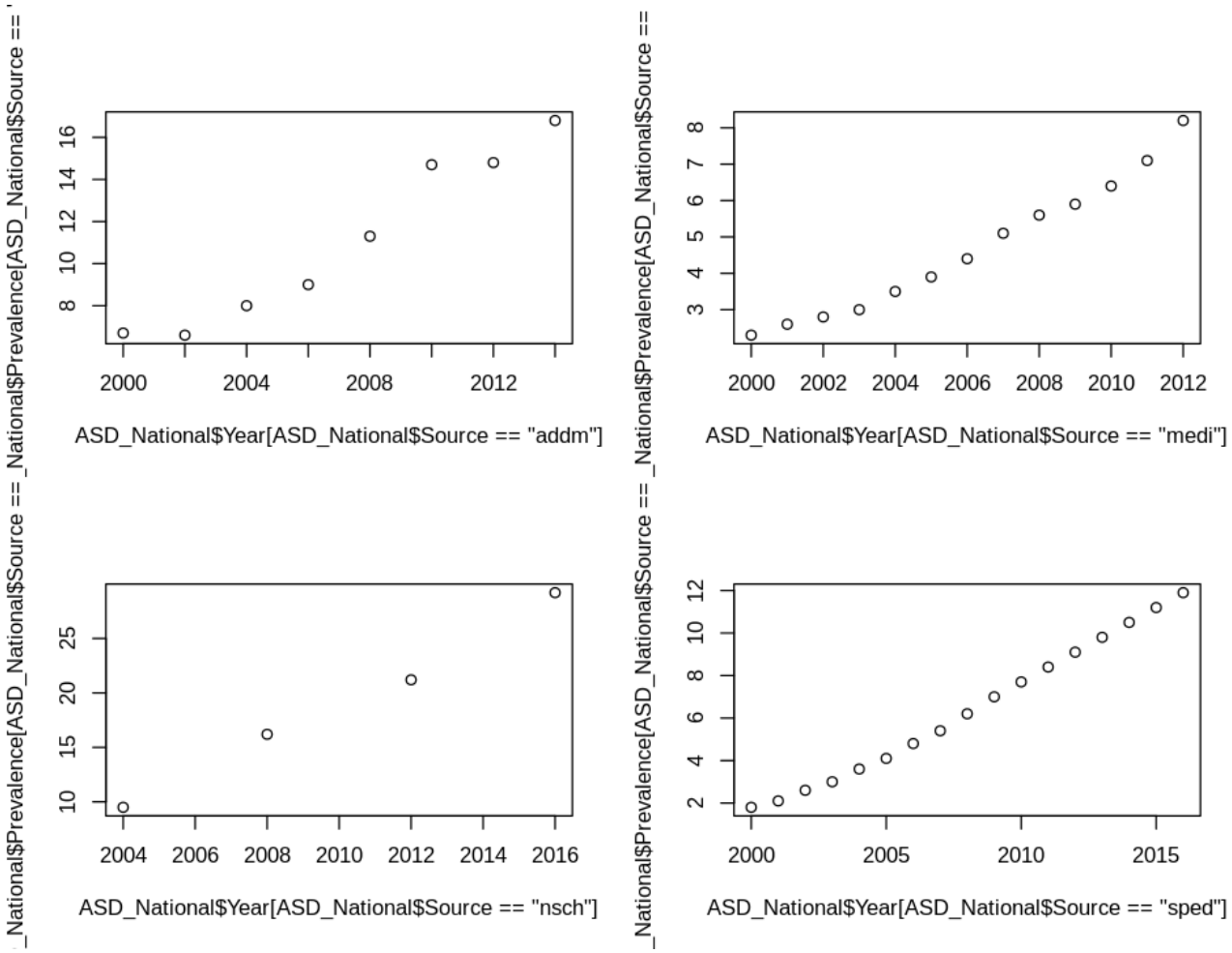
In [119]:
```
# Use Year_factor as x-axis: y value Prevalence is     aggregated for differen
plot(ASD_National$Year_Factor, ASD_National$Prevalence)
```



In [120]:
```
# table(ASD_National$Source_Full3)
```

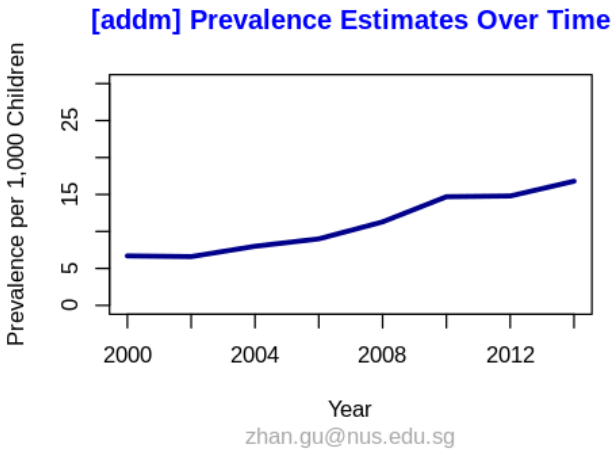
In [121]:

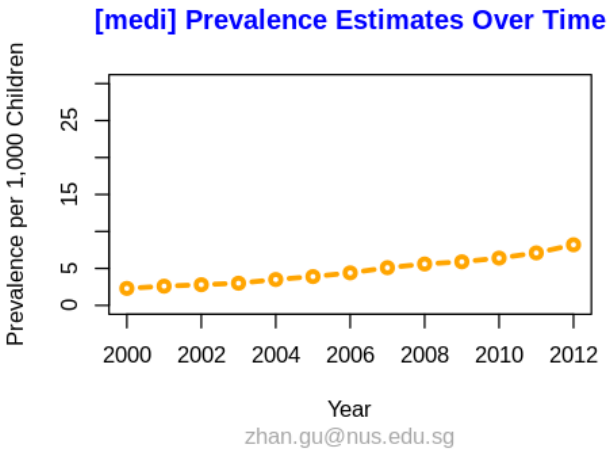
```
# Adjust in-line plot size to M x N
options(repr.plot.width=8, repr.plot.height=6)

par(mfrow=c(2, 2))

# Prevalence over Year, from data source:
# addm-Autism & Developmental Disabilities Monitoring Network
plot(ASD_National$Year[ASD_National$Source == 'addm'],
     ASD_National$Prevalence[ASD_National$Source == 'addm'])

# Prevalence over Year, from data source:
# medi-Medicaid
plot(ASD_National$Year[ASD_National$Source == 'medi'],
     ASD_National$Prevalence[ASD_National$Source == 'medi'])

# Prevalence over Year, from data source:
# nsch-National Survey of Children Health
plot(ASD_National$Year[ASD_National$Source == 'nsch'],
     ASD_National$Prevalence[ASD_National$Source == 'nsch'])

# Prevalence over Year, from data source:
# sped-Special Education Child Count
plot(ASD_National$Year[ASD_National$Source == 'sped'],
     ASD_National$Prevalence[ASD_National$Source == 'sped'])

par(mfrow=c(1, 1)) # Reset to one plot on one page
```

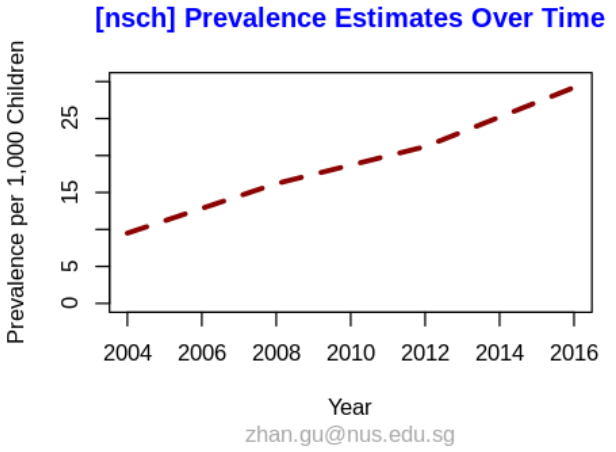


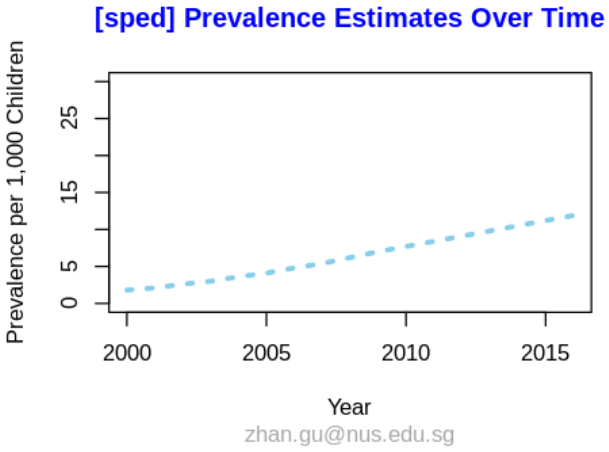
In [122]:

```
# ---------------------------------
# Add more annotations to above plots
# ---------------------------------
# Color list
# addm : darkblue
# medi : orange
# nsch : darkred
# sped : skyblue

par(mfrow=c(2, 2))

# Prevalence over Year, from data source:
# addm-Autism & Developmental Disabilities Monitoring Network
plot(ASD_National$Year[ASD_National$Source == 'addm'],
     ASD_National$Prevalence[ASD_National$Source == 'addm'],
     type="l", # dot/point type
     lty=1, # line type
     lwd=3, # line width
     col="darkblue", # line color
     xlab="Year",
     ylab="Prevalence per 1,000 Children",
     ylim = c(0, 30), # Set value range of y axis
     main="[addm] Prevalence Estimates Over Time",
     sub  = "zhan.gu@nus.edu.sg",
     col.main="blue", col.lab="black", col.sub="darkgrey")

# Prevalence over Year, from data source:
# medi-Medicaid
plot(ASD_National$Year[ASD_National$Source == 'medi'],
     ASD_National$Prevalence[ASD_National$Source == 'medi'],
     type="b", lty=1, lwd=3,  col="orange",
     xlab="Year",
     ylab="Prevalence per 1,000 Children",
     ylim = c(0, 30), # Set value range of y axis
     main="[medi] Prevalence Estimates Over Time",
     sub  = "zhan.gu@nus.edu.sg",
     col.main="blue", col.lab="black", col.sub="darkgrey")

# Prevalence over Year, from data source:
# nsch-National Survey of Children Health
plot(ASD_National$Year[ASD_National$Source == 'nsch'],
     ASD_National$Prevalence[ASD_National$Source == 'nsch'],
     type="l", lty=2, lwd=3,  col="darkred",
     xlab="Year",
     ylab="Prevalence per 1,000 Children",
     ylim = c(0, 30), # Set value range of y axis
     main="[nsch] Prevalence Estimates Over Time",
     sub  = "zhan.gu@nus.edu.sg",
     col.main="blue", col.lab="black", col.sub="darkgrey")

# Prevalence over Year, from data source:
# sped-Special Education Child Count
plot(ASD_National$Year[ASD_National$Source == 'sped'],
     ASD_National$Prevalence[ASD_National$Source == 'sped'],
     type="l", lty=3, lwd=3,  col="skyblue",
     xlab="Year",
     ylab="Prevalence per 1,000 Children",
     ylim = c(0, 30), # Set value range of y axis
     main="[sped] Prevalence Estimates Over Time",
     sub  = "zhan.gu@nus.edu.sg",
     col.main="blue", col.lab="black", col.sub="darkgrey")

par(mfrow=c(1, 1)) # Reset to one plot on one page
```







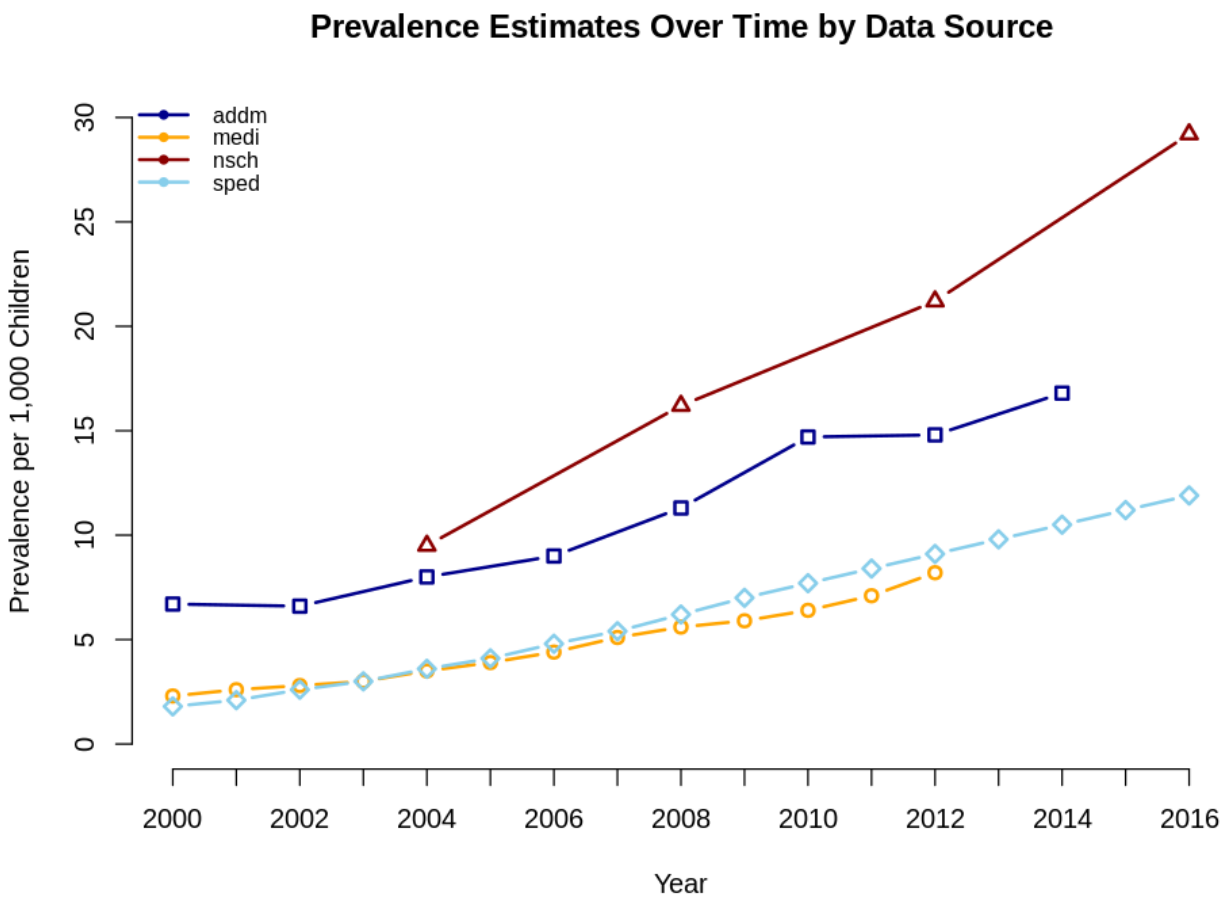


Data Visualisation (Base Graphic) - [R] REPORTED PREVALENCE HAS CHANGED OVER TIME by [Data Source]

Create multiple lines within a single chart

In [123]:

```r
# ---------------------------------
# [National] < Prevalence Varies over Time/Year by Data Source >
# ---------------------------------
# Create a first line
plot(ASD_National$Year[ASD_National$Source == 'addm'],
     ASD_National$Prevalence[ASD_National$Source == 'addm'],
     col = "darkblue", lty = 1, lwd = 2,
     type = "b", # use dot/point
     pch = 0, # dot/point type: http://www.endmemo.com/program/R/pchsymbols.ph
     xlab="Year",
     xlim=c(2000, 2016), # Set x axis value range
     ylab="Prevalence per 1,000 Children",
     ylim=c(0, 30), # Set y axis value range
     main="Prevalence Estimates Over Time by Data Source",
     col.main="black", col.lab="black", col.sub="grey",
     frame = FALSE, # Remove frame
     axes=FALSE # Remove x and y axis
)
axis(1, at=seq(2000, 2016, 1)) # Customize x axis
axis(2, at=seq(0, 30, 5)) # Customize y axis

# Add another line
lines(ASD_National$Year[ASD_National$Source == 'medi'],
      ASD_National$Prevalence[ASD_National$Source == 'medi'],
      pch = 1, col = "orange", type = "b", lty = 1, lwd = 2
)
# Add another line
lines(ASD_National$Year[ASD_National$Source == 'nsch'],
      ASD_National$Prevalence[ASD_National$Source == 'nsch'],
      pch = 2, col = "darkred", type = "b", lty = 1, lwd = 2
)
# Add another line
lines(ASD_National$Year[ASD_National$Source == 'sped'],
      ASD_National$Prevalence[ASD_National$Source == 'sped'],
      pch = 5, col = "skyblue", type = "b", lty = 1, lwd = 2
)
# Add a legend to the plot
legend("topleft", legend=levels(ASD_National$Source),
       col=c("darkblue", "orange", "darkred", "skyblue"),
       pch = 20, # dot in a line
       lty = 1, # line type
       lwd = 2, # line width
       cex=0.8, # size of text
       bty = 'n' # Without frame
)
```



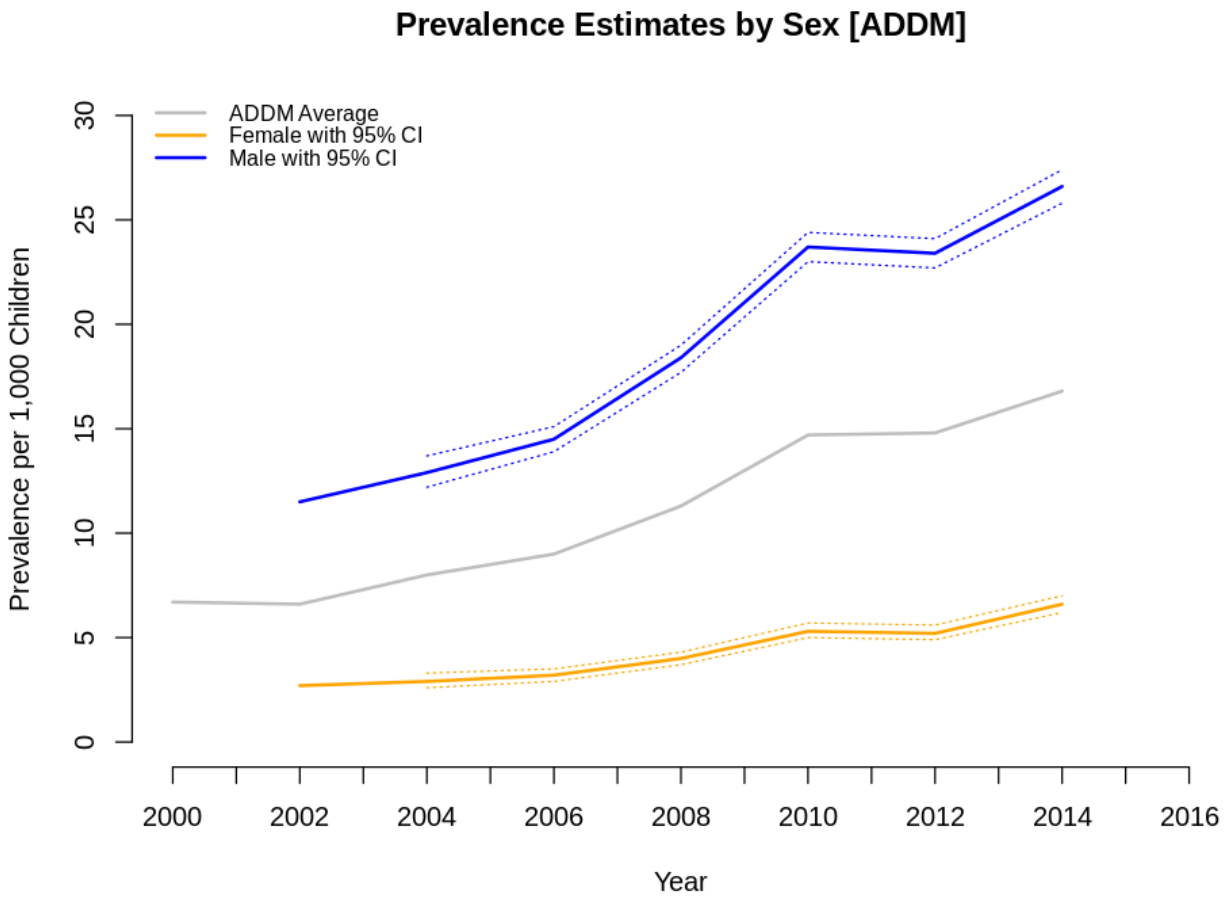
R pch: dot/point type: http://www.endmemo.com/program/R/pchsymbols.php (http://www.endmemo.com/program/R/pchsymbols.php)

R plot colour list: https://www.r-graph-gallery.com/42-colors-names.html (https://www.r-graph-gallery.com/42-colors-names.html)

Data Visualisation (Base Graphic) - [R] REPORTED PREVALENCE VARIES BY SEX [Source: ADDM] over [Year]

In [124]:

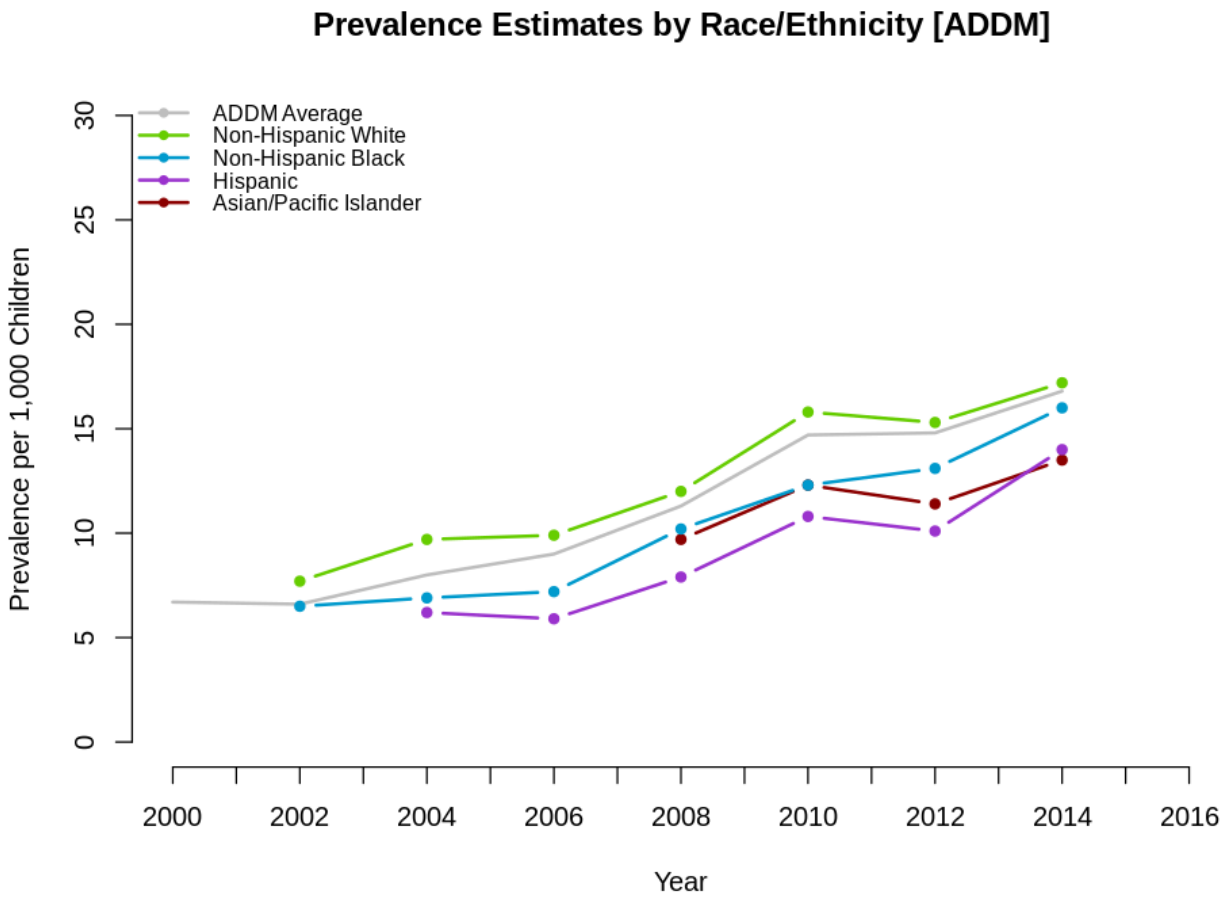
```
# ---------------------------------
# [addm] < Prevalence Varies by Sex >
# ---------------------------------
# Create a first line
plot(ASD_National$Year[ASD_National$Source == 'addm'],
     ASD_National$Prevalence[ASD_National$Source == 'addm'],
     col = "grey", lty = 1, lwd = 2,
     type = "l", # use dot/point
     pch = 0, # dot/point type: http://www.endmemo.com/program/R/pchsymbols.ph
     xlab="Year",
     xlim=c(2000, 2016), # Set x axis value range
     ylab="Prevalence per 1,000 Children",
     ylim=c(0, 30), # Set y axis value range
     main="Prevalence Estimates by Sex [ADDM]",
     col.main="black", col.lab="black", col.sub="grey",
     frame = FALSE, # Remove frame
     axes=FALSE # Remove x and y axis
)
axis(1, at=seq(2000, 2016, 1)) # Customize x axis
axis(2, at=seq(0, 30, 5)) # Customize y axis

# Add Female prevalence
lines(ASD_National$Year[ASD_National$Source == 'addm'],
      ASD_National$Female.Prevalence[ASD_National$Source == 'addm'],
      pch = 1, col = "orange", type = "l", lty = 1, lwd = 2)
# Add Female prevalence lower CI
lines(ASD_National$Year[ASD_National$Source == 'addm'],
      ASD_National$Female.Lower.CI[ASD_National$Source == 'addm'],
      pch = 1, col = "orange", type = "l", lty = 3, lwd = 1)
# Add Female prevalence upper CI
lines(ASD_National$Year[ASD_National$Source == 'addm'],
      ASD_National$Female.Upper.CI[ASD_National$Source == 'addm'],
      pch = 1, col = "orange", type = "l", lty = 3, lwd = 1)

# Add Male prevalence
lines(ASD_National$Year[ASD_National$Source == 'addm'],
      ASD_National$Male.Prevalence[ASD_National$Source == 'addm'],
      pch = 1, col = "blue", type = "l", lty = 1, lwd = 2)
# Add Male prevalence lower CI
lines(ASD_National$Year[ASD_National$Source == 'addm'],
      ASD_National$Male.Lower.CI[ASD_National$Source == 'addm'],
      pch = 1, col = "blue", type = "l", lty = 3, lwd = 1)
# Add Male prevalence upper CI
lines(ASD_National$Year[ASD_National$Source == 'addm'],
      ASD_National$Male.Upper.CI[ASD_National$Source == 'addm'],
      pch = 1, col = "blue", type = "l", lty = 3, lwd = 1)
# Add a legend to the plot
legend("topleft", legend=c('ADDM Average', 'Female with 95% CI', 'Male with 95
       col=c("grey", "orange", "blue"),
       #       pch = 20, # dot in a line
       lty = 1, # line type
       lwd = 2, # line width
       cex=0.8, # size of text
       bty = 'n' # Without frame
)
```



Data Visualisation (Base Graphic) - [R] REPORTED PREVALENCE VARIES BY RACE AND ETHNICITY [Source: ADDM]

In [125]:

```
# ---------------------------------
# [addm] < Prevalence Varies by Race and Ethnicity >
# ---------------------------------
# Create a first line
plot(ASD_National$Year[ASD_National$Source == 'addm'],
     ASD_National$Prevalence[ASD_National$Source == 'addm'],
     col = "grey", lty = 1, lwd = 2,
     type = "l", # use dot/point
     pch = 0, # dot/point type: http://www.endmemo.com/program/R/pchsymbols.ph
     xlab="Year",
     xlim=c(2000, 2016), # Set x axis value range
     ylab="Prevalence per 1,000 Children",
     ylim=c(0, 30), # Set y axis value range
     main="Prevalence Estimates by Race/Ethnicity [ADDM]",
     col.main="black", col.lab="black", col.sub="grey",
     frame = FALSE, # Remove frame
     axes=FALSE # Remove x and y axis
)
axis(1, at=seq(2000, 2016, 1)) # Customize x axis
axis(2, at=seq(0, 30, 5)) # Customize y axis

# R plot colour list: https://www.r-graph-gallery.com/42-colors-names.html

# Add Asian.or.Pacific.Islander.Prevalence
lines(ASD_National$Year[ASD_National$Source == 'addm'],
      ASD_National$Asian.or.Pacific.Islander.Prevalence[ASD_National$Source ==
      pch = 20, col = "darkred", type = "b", lty = 1, lwd = 2)
# Add Hispanic.Prevalence
lines(ASD_National$Year[ASD_National$Source == 'addm'],
      ASD_National$Hispanic.Prevalence[ASD_National$Source == 'addm'],
      pch = 20, col = "darkorchid3", type = "b", lty = 1, lwd = 2)
# Add Non.hispanic.black.Prevalence
lines(ASD_National$Year[ASD_National$Source == 'addm'],
      ASD_National$Non.hispanic.black.Prevalence[ASD_National$Source == 'addm'
      pch = 20, col = "deepskyblue3", type = "b", lty = 1, lwd = 2)
# Add Non.hispanic.white.Prevalence
lines(ASD_National$Year[ASD_National$Source == 'addm'],
      ASD_National$Non.hispanic.white.Prevalence[ASD_National$Source == 'addm'
      pch = 20, col = "chartreuse3", type = "b", lty = 1, lwd = 2)

# Add a legend to the plot
legend("topleft", legend=c('ADDM Average',
                           'Non-Hispanic White',
                           'Non-Hispanic Black',
                           'Hispanic',
                           'Asian/Pacific Islander'),
       col=c("grey", "chartreuse3", "deepskyblue3", "darkorchid3", "darkred"),
       pch = 20, # dot in a line
       lty = 1, # line type
       lwd = 2, # line width
       cex=0.8, # size of text
       bty = 'n' # Without frame
)
```



In [126]:
```
# Adjust in-line plot size to M x N
options(repr.plot.width=8, repr.plot.height=4)
```

Quiz:

Add 95% Confidence Interval to above plot

In [127]:
```
# Write your code below and press Shift+Enter to execute
```

Double-click **here** for the solution.

Quiz:

Use talbe() to count No. prevalence records for each Data Source. Then use barplot() to visualize.

In [128]:
```
# Write your code below and press Shift+Enter to execute
```

Double-click **here** for the solution.

Quiz:

Which Data Sources are available in which years?

In [129]:
```
# Write your code below and press Shift+Enter to execute
```

Double-click **here** for the solution.

Quiz:

Which Data Source has breakdown Prevalvence data by sex/gender?

In [130]:
```
# Write your code below and press Shift+Enter to execute
```

Double-click **here** for the solution.

Quiz:

Which Data Source has breakdown Prevalvence data by race and ethnicity?

In [131]:
```
# Write your code below and press Shift+Enter to execute
```

Double-click **here** for the solution.

Data Visualisation (Enhanced)

In [132]:
```
if(!require(ggplot2)){install.packages("ggplot2")}
library(ggplot2)
```

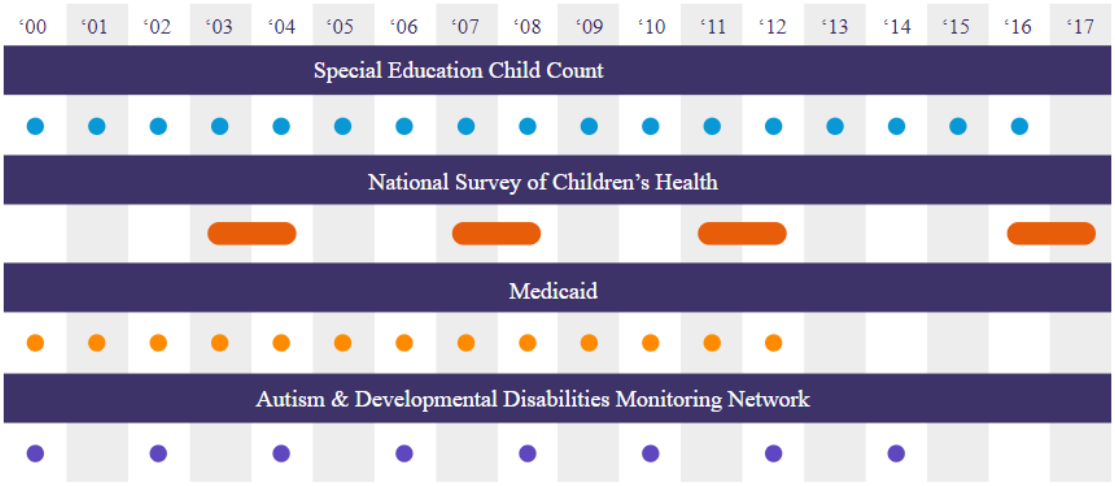
```
Loading required package: ggplot2
```

In [133]:
```
# Adjust in-line plot size to M x N
options(repr.plot.width=8, repr.plot.height=4)
```

Data Visualisation (Enhanced) - [CDC] Explore the Data

Years Data Available

Select state: U.S. or Total†



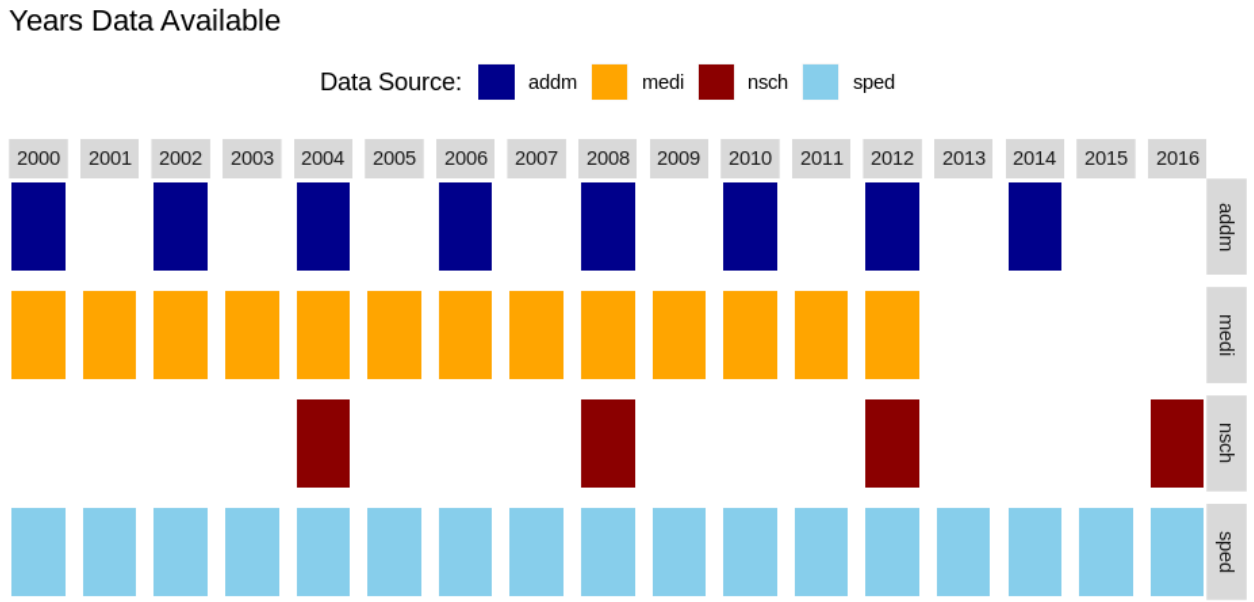
WHY THIS MATTERS

Because ASD data are collected at specific times, they provide a snapshot of what was going on at a certain moment in time. Findings from different data sources are typically reported a year or more *after* the data were collected; therefore, prevalence may have changed between the time data were collected and the time they were reported.

**ADDM estimate = the total for all sites combined.*

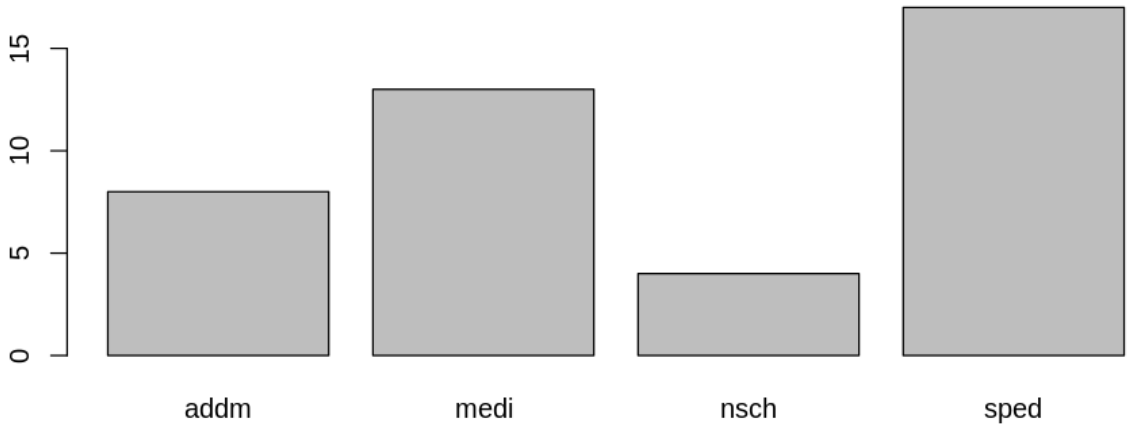
Data Visualisation (Enhanced) - [R] Explore the Data

In [134]:
```
# ---------------------------------
# [National] < Years Data Available >
# ---------------------------------
p = ggplot(ASD_National, aes(x = 1, fill = Source)) +
  geom_bar() + theme(axis.text.x=element_blank(),  # Hide axis
                     axis.ticks.x=element_blank(), # Hide axis
                     axis.text.y=element_blank(),  # Hide axis
                     axis.ticks.y=element_blank(), # Hide axis
                     panel.background = element_blank(), # Remove panel backgr
                     legend.position="top"
  ) +
  scale_fill_manual("Data Source:", values = c("addm" = "darkblue",
                                               "medi" = "orange",
                                               "nsch" = "darkred",
                                               "sped" = "skyblue")) +
  labs(x="", y="", title="Years Data Available") + # layers of graphics
  facet_grid(facets = Source~Year)
# Show plot
p
```

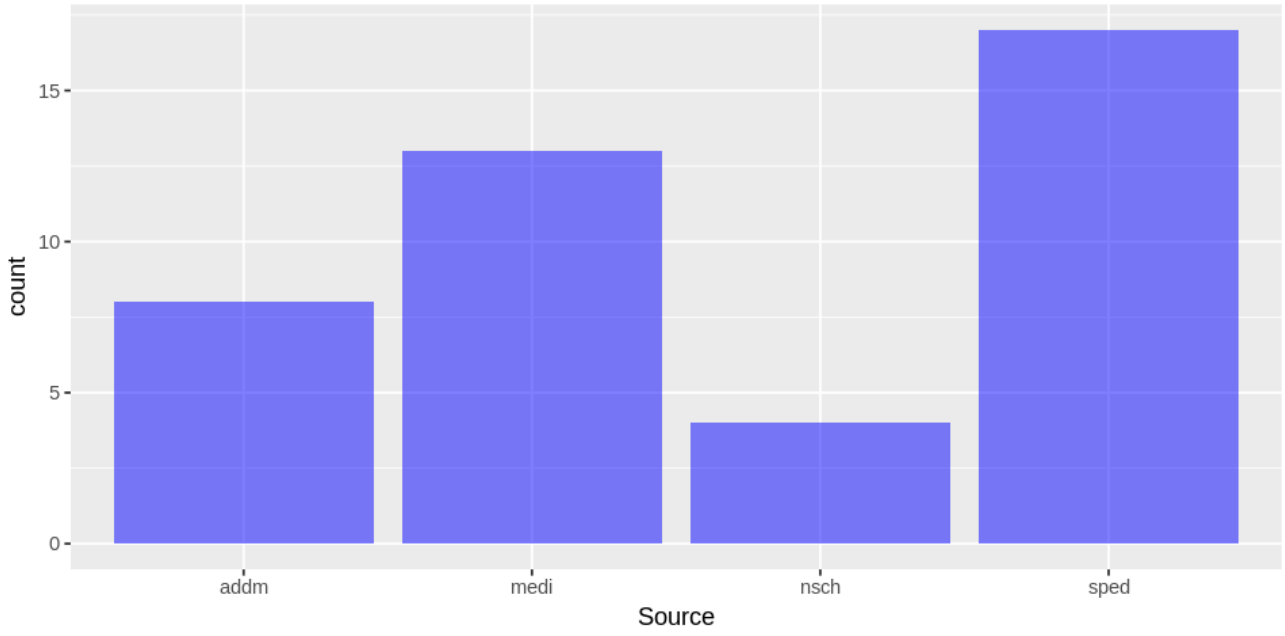


Data Visualisation (Enhanced) - Barplot

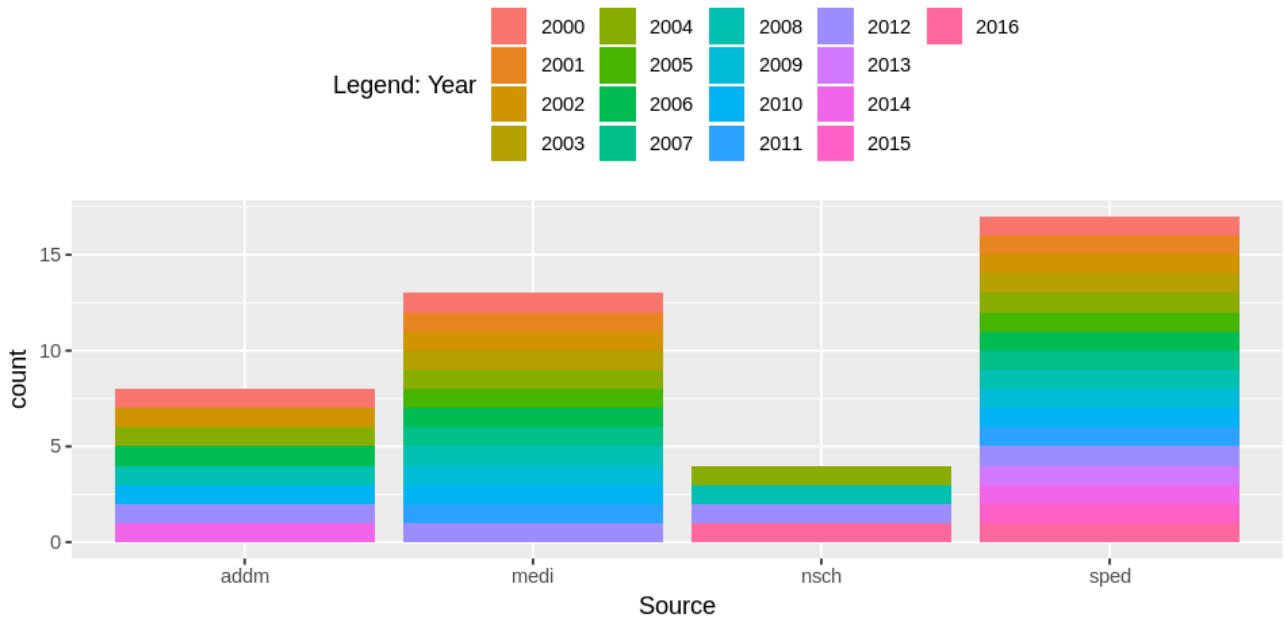
In [135]:
```
# Create bar chart using R graphics
barplot(table(ASD_National$Source))
```



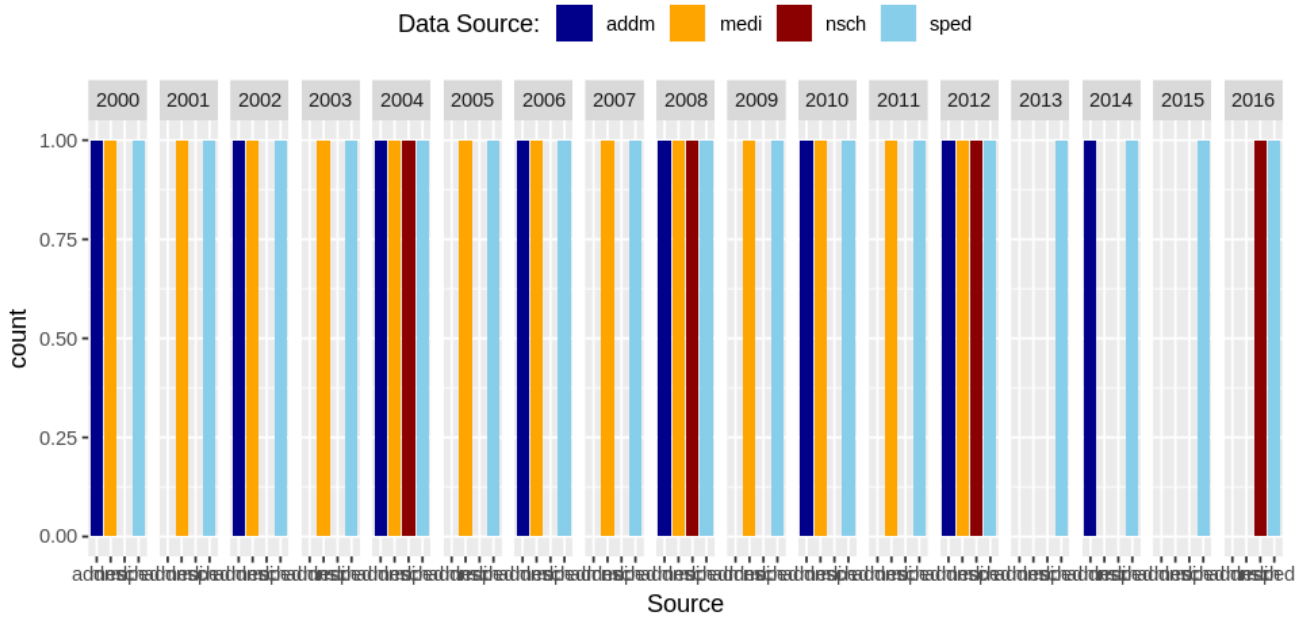
In [136]:
```
# Create bar chart using ggplot2
ggplot(ASD_National, aes(x = Source)) + geom_bar(fill = "blue", alpha=0.5)
```



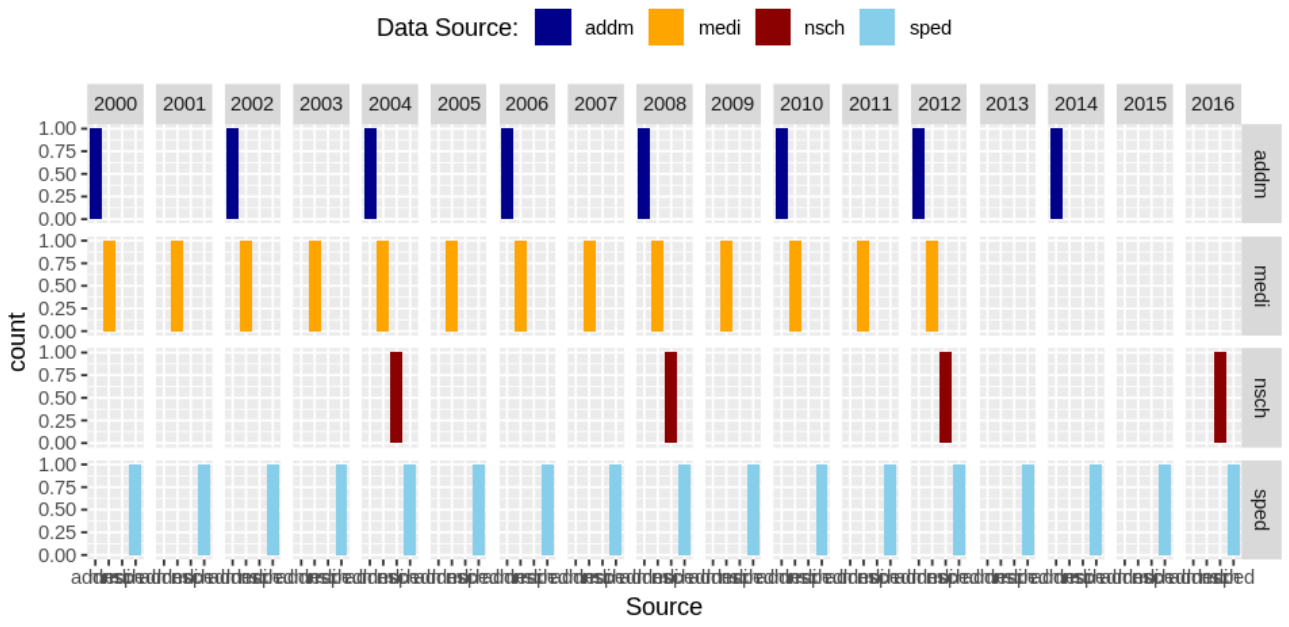
In [137]:
```
# Use color to differentiate sub-group data (Year)
ggplot(ASD_National, aes(x = Source, fill = factor(Year))) + geom_bar() +
  theme(legend.position="top") + labs(fill = "Legend: Year")
```



In [138]:
```
# Split chart to mutiple columns by using: facets = . ~ Year
ggplot(ASD_National, aes(x = Source, fill = Source)) + geom_bar() +
  theme(legend.position="top") +
  scale_fill_manual("Data Source:", values = c("addm" = "darkblue",
                                               "medi" = "orange",
                                               "nsch" = "darkred",
                                               "sped" = "skyblue")) +

  facet_grid(facets = . ~ Year)
```



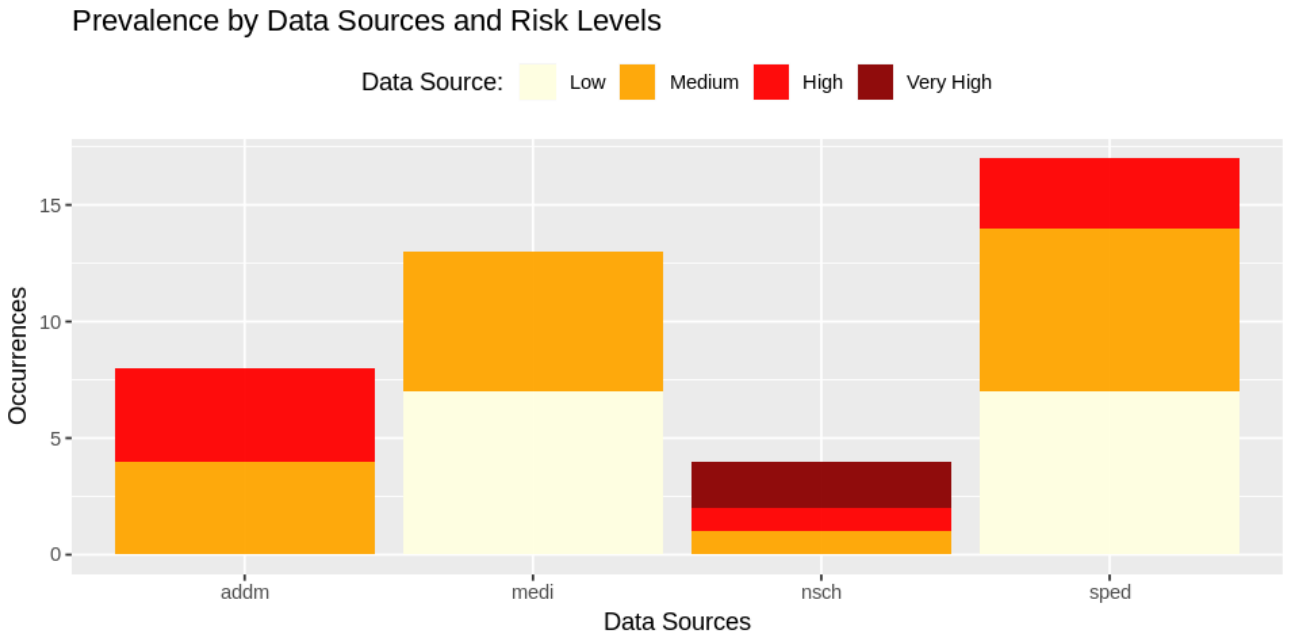
In [139]:
```
# Split chart to mutiple rows and columns by using: facets = Source ~ Year
ggplot(ASD_National, aes(x = Source, fill = Source)) + geom_bar() +
  theme(legend.position="top") +
  scale_fill_manual("Data Source:", values = c("addm" = "darkblue",
                                               "medi" = "orange",
                                               "nsch" = "darkred",
                                               "sped" = "skyblue")) +
  facet_grid(facets = Source~Year)
```



Above chart is now very similar to earlier [National] < Years Data Available >.

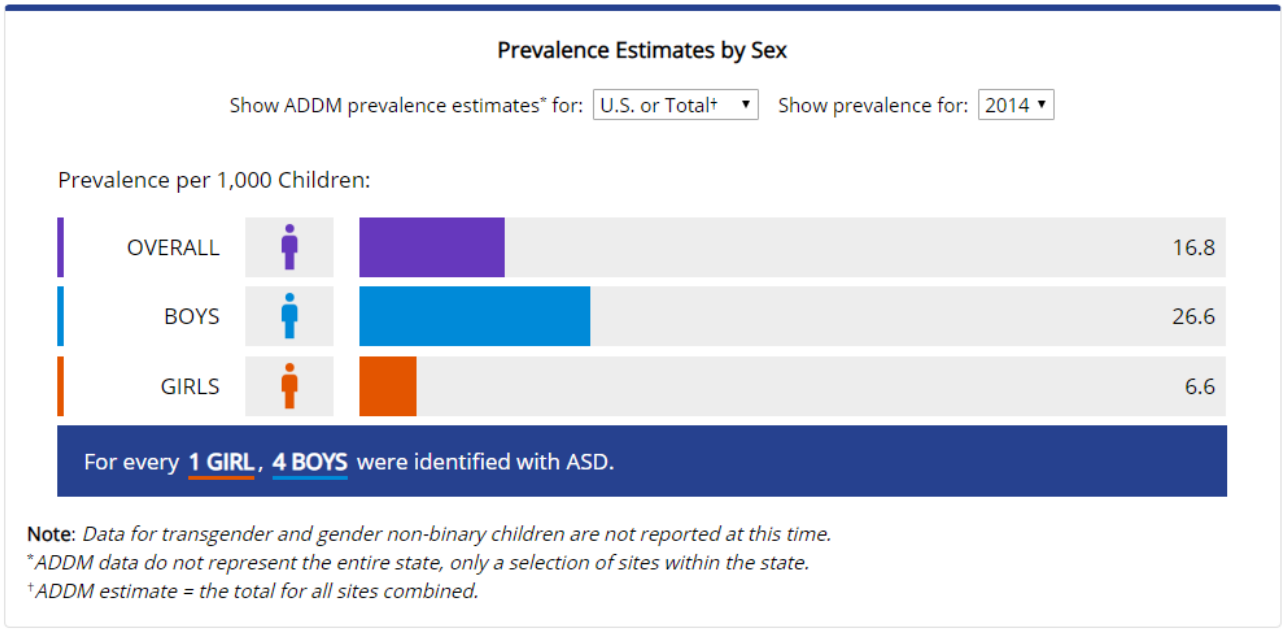
Data Visualisation (Enhanced) - [R] Prevalence by Data Sources and Risk Levels

In [140]:
```
# Use color to differentiate sub-group data (Year)
ggplot(ASD_National, aes(x = Source, fill = Prevalence_Risk4)) +
  geom_bar(alpha=0.95, position = position_stack(reverse = TRUE)) + # Reverse
  scale_fill_manual("Data Source:", values = c("Low" = "lightyellow",
                                               "Medium" = "orange",
                                               "High" = "red",
                                               "Very High" = "darkred")) +
  labs(x="Data Sources", y="Occurrences", title="Prevalence by Data Sources an
  theme(legend.position="top") + labs(fill = "Legend: Risk")
```



Barplot / Column plot

Data Visualisation (Enhanced) - [CDC] REPORTED PREVALENCE VARIES BY SEX



Data Visualisation (Enhanced) - [R] REPORTED PREVALENCE VARIES BY SEX [Source: ADDM] [Year: 2014]

In [141]:
```
# Filter only data of ADDM
ASD_National_ADDM <- subset(ASD_National, Source == 'addm')
#
ASD_National_ADDM
```

Source	Year	Prevalence	Upper.CI	Lower.CI	Source_Full1	Source_Full2	Male.Prevalence	Male.Lower
addm	2000	6.7	7.0	6.3	Autism & Developmental Disabilities Monitoring Network	addm-Autism & Developmental Disabilities Monitoring Network	NA	
addm	2002	6.6	6.8	6.3	Autism & Developmental Disabilities Monitoring Network	addm-Autism & Developmental Disabilities Monitoring Network	11.5	
addm	2004	8.0	8.4	7.6	Autism & Developmental Disabilities Monitoring Network	addm-Autism & Developmental Disabilities Monitoring Network	12.9	1
addm	2006	9.0	9.3	8.6	Autism & Developmental Disabilities Monitoring Network	addm-Autism & Developmental Disabilities Monitoring Network	14.5	1
addm	2008	11.3	11.7	11.0	Autism & Developmental Disabilities Monitoring Network	addm-Autism & Developmental Disabilities Monitoring Network	18.4	1
addm	2010	14.7	15.1	14.3	Autism & Developmental Disabilities Monitoring Network	addm-Autism & Developmental Disabilities Monitoring Network	23.7	2
addm	2012	14.8	15.2	14.4	Autism & Developmental Disabilities Monitoring Network	addm-Autism & Developmental Disabilities Monitoring Network	23.4	2
addm	2014	16.8	17.3	16.4	Autism & Developmental Disabilities Monitoring Network	addm-Autism & Developmental Disabilities Monitoring Network	26.6	2

In [142]:
```
# Construct a new re-shaped dataframe of [ Source: ADDM ] [Year: 2014]
#
Process_Source = 'addm'
Process_Year = 2014
```

Define a function to create a re-shaped dataframe:

In [143]:
```
Function_Reshape_ASD_National_ADDM <- function(Process_Source, Process_Year) {
    # Create the vectors:
    Sex.Group  = c('Overall',
                   'Boys',
                   'Girls')
    Sex.Group

    Prevalence = c(ASD_National_ADDM$Prevalence[ASD_National_ADDM$Year == Proc
                   ASD_National_ADDM$Male.Prevalence[ASD_National_ADDM$Year ==
                   ASD_National_ADDM$Female.Prevalence[ASD_National_ADDM$Year
    Prevalence

    # Combine all the vectors into a data frame:
    ASD_National_ADDM_Reshaped_DF = data.frame(Sex.Group, Prevalence, stringsA

    # Add new columns:
    ASD_National_ADDM_Reshaped_DF$Source = Process_Source
    ASD_National_ADDM_Reshaped_DF$Year = Process_Year
    return(ASD_National_ADDM_Reshaped_DF) # Return a dataframe
}
```

Use defined function Function_Reshape_ASD_National_ADDM( ) for a specific year:

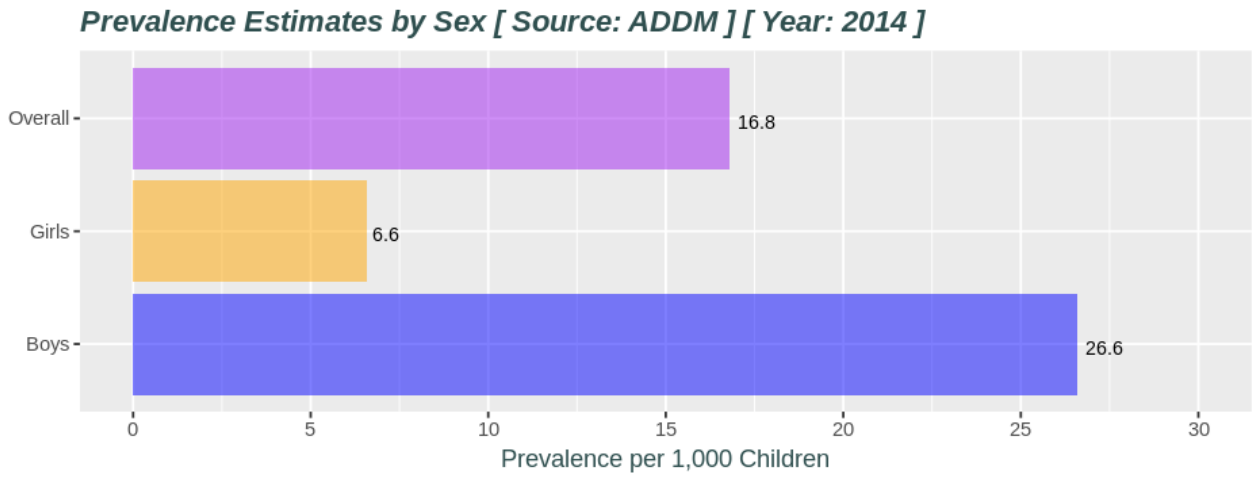
In [144]:
```
ASD_National_ADDM_Reshaped_DF <- Function_Reshape_ASD_National_ADDM(Process_So
ASD_National_ADDM_Reshaped_DF
```

Sex.Group	Prevalence	Source	Year
Overall	16.8	addm	2014
Boys	26.6	addm	2014
Girls	6.6	addm	2014

Visualise: **Prevalence Estimates by Sex [Source: ADDM] [Year: 2014]**

In [145]:
```
# Adjust in-line plot size to M x N
options(repr.plot.width=8, repr.plot.height=3)
```

In [146]:
```
ggplot(ASD_National_ADDM_Reshaped_DF, aes(Sex.Group, Prevalence)) +
  geom_col(aes(fill = Sex.Group, colours = ), alpha=0.5) + # Use column chart
  geom_text(aes(label = Prevalence), vjust = +0.75, hjust = -0.2, size = 3) +
  scale_y_continuous(name = "Prevalence per 1,000 Children",
                     breaks = seq(0, 30, 5),
                     limits=c(0, 30)) +
  scale_x_discrete(name = "") +
  scale_fill_manual("Sex Group:", values = c("Overall" = "purple",
                                             "Boys" = "blue",
                                             "Girls" = "orange")) +
  ggtitle("Prevalence Estimates by Sex [ Source: ADDM ] [ Year: 2014 ]") +
  theme(title = element_text(face = 'bold.italic', color = "darkslategrey"),
        axis.title = element_text(face = 'plain', color = "darkslategrey"),
        legend.position = 'none') +
  coord_flip()  # Rotate chart
#  facet_grid(facets = Year ~ .)
```



Data Visualisation (Enhanced) - [R] REPORTED PREVALENCE VARIES BY SEX [Source: ADDM] [Year: ALL]

In [147]:
```
# Create a new datafarme to hold re-shaped data for all years.
ASD_National_ADDM_Reshaped_DF_All = ASD_National_ADDM_Reshaped_DF # Loaded wit
```

In [148]:
```
Process_Source = 'addm'
unique(ASD_National_ADDM$Year)
```

2000 2002 2004 2006 2008 2010 2012 2014

Use defined function Function_Reshape_ASD_National_ADDM( ) for ALL remaining years:

In [149]:
```
ASD_National_ADDM_Reshaped_DF <- Function_Reshape_ASD_National_ADDM(Process_Sc
ASD_National_ADDM_Reshaped_DF
# Append rows to existing dataframe, using Row Bind function: rbind()
ASD_National_ADDM_Reshaped_DF_All = rbind(ASD_National_ADDM_Reshaped_DF_All, A
```

Sex.Group	Prevalence	Source	Year
Overall	14.8	addm	2012
Boys	23.4	addm	2012
Girls	5.2	addm	2012

In [150]:
```
ASD_National_ADDM_Reshaped_DF <- Function_Reshape_ASD_National_ADDM(Process_Sc
ASD_National_ADDM_Reshaped_DF
# Append rows to existing dataframe, using Row Bind function: rbind()
ASD_National_ADDM_Reshaped_DF_All = rbind(ASD_National_ADDM_Reshaped_DF_All, A
```

Sex.Group	Prevalence	Source	Year
Overall	14.7	addm	2010
Boys	23.7	addm	2010
Girls	5.3	addm	2010

In [151]:
```
ASD_National_ADDM_Reshaped_DF <- Function_Reshape_ASD_National_ADDM(Process_Sc
ASD_National_ADDM_Reshaped_DF
# Append rows to existing dataframe, using Row Bind function: rbind()
ASD_National_ADDM_Reshaped_DF_All = rbind(ASD_National_ADDM_Reshaped_DF_All, A
```

Sex.Group	Prevalence	Source	Year
Overall	11.3	addm	2008
Boys	18.4	addm	2008
Girls	4.0	addm	2008

In [152]:
```
ASD_National_ADDM_Reshaped_DF <- Function_Reshape_ASD_National_ADDM(Process_Sc
ASD_National_ADDM_Reshaped_DF
# Append rows to existing dataframe, using Row Bind function: rbind()
ASD_National_ADDM_Reshaped_DF_All = rbind(ASD_National_ADDM_Reshaped_DF_All, A
```

Sex.Group	Prevalence	Source	Year
Overall	9.0	addm	2006
Boys	14.5	addm	2006
Girls	3.2	addm	2006

In [153]:
```
ASD_National_ADDM_Reshaped_DF <- Function_Reshape_ASD_National_ADDM(Process_Sc
ASD_National_ADDM_Reshaped_DF
# Append rows to existing dataframe, using Row Bind function: rbind()
ASD_National_ADDM_Reshaped_DF_All = rbind(ASD_National_ADDM_Reshaped_DF_All, A
```

Sex.Group	Prevalence	Source	Year
Overall	8.0	addm	2004
Boys	12.9	addm	2004
Girls	2.9	addm	2004

In [154]:
```
ASD_National_ADDM_Reshaped_DF <- Function_Reshape_ASD_National_ADDM(Process_Sc
ASD_National_ADDM_Reshaped_DF
# Append rows to existing dataframe, using Row Bind function: rbind()
ASD_National_ADDM_Reshaped_DF_All = rbind(ASD_National_ADDM_Reshaped_DF_All, A
```

Sex.Group	Prevalence	Source	Year
Overall	6.6	addm	2002
Boys	11.5	addm	2002
Girls	2.7	addm	2002

In [155]:
```
ASD_National_ADDM_Reshaped_DF <- Function_Reshape_ASD_National_ADDM(Process_Sc
ASD_National_ADDM_Reshaped_DF
# Append rows to existing dataframe, using Row Bind function: rbind()
ASD_National_ADDM_Reshaped_DF_All = rbind(ASD_National_ADDM_Reshaped_DF_All, A
```

Sex.Group	Prevalence	Source	Year
Overall	6.7	addm	2000
Boys	NA	addm	2000
Girls	NA	addm	2000

In [156]:
```
# Re-shaped ADDM data for ALL years:
ASD_National_ADDM_Reshaped_DF_All
```

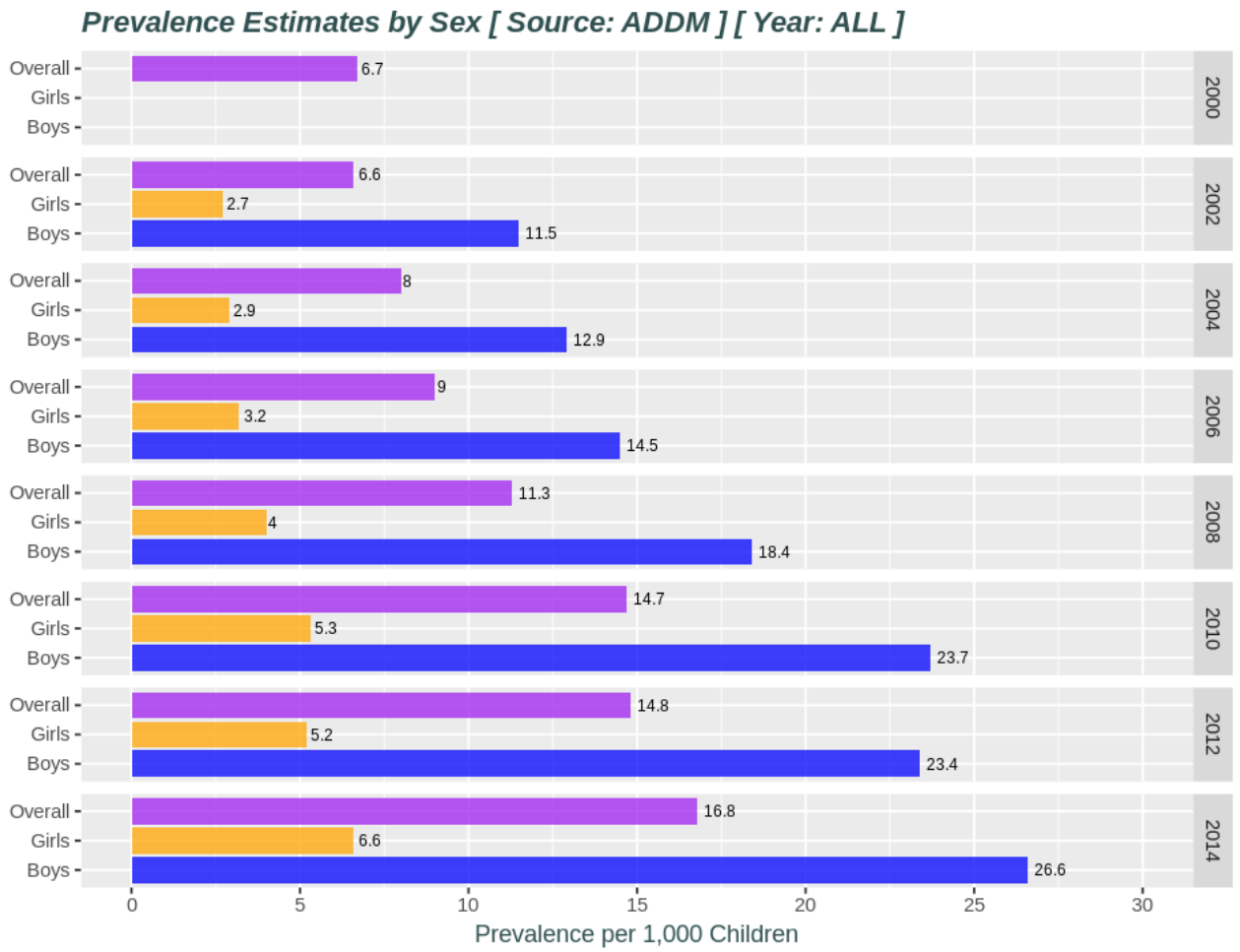
Sex.Group	Prevalence	Source	Year
Overall	16.8	addm	2014
Boys	26.6	addm	2014
Girls	6.6	addm	2014
Overall	14.8	addm	2012
Boys	23.4	addm	2012
Girls	5.2	addm	2012
Overall	14.7	addm	2010
Boys	23.7	addm	2010
Girls	5.3	addm	2010
Overall	11.3	addm	2008
Boys	18.4	addm	2008
Girls	4.0	addm	2008
Overall	9.0	addm	2006
Boys	14.5	addm	2006
Girls	3.2	addm	2006
Overall	8.0	addm	2004
Boys	12.9	addm	2004
Girls	2.9	addm	2004
Overall	6.6	addm	2002
Boys	11.5	addm	2002
Girls	2.7	addm	2002
Overall	6.7	addm	2000
Boys	NA	addm	2000
Girls	NA	addm	2000

Visualise: **Prevalence Estimates by Sex [Source: ADDM] [Year: ALL]**

In [157]:
```
# Adjust in-line plot size to M x N
options(repr.plot.width=8, repr.plot.height=6)
```

In [158]:
```
ggplot(ASD_National_ADDM_Reshaped_DF_All, aes(Sex.Group, Prevalence)) +
  geom_col(aes(fill = Sex.Group, colours = ), alpha=0.75) + # Use column chart
  geom_text(aes(label = Prevalence), vjust = +0.5, hjust = -0.2, size = 2.5) +
  scale_y_continuous(name = "Prevalence per 1,000 Children",
                     breaks = seq(0, 30, 5),
                     limits=c(0, 30)) +
  scale_x_discrete(name = "") +
  scale_fill_manual("Sex Group:", values = c("Overall" = "purple",
                                             "Boys" = "blue",
                                             "Girls" = "orange")) +
  ggtitle("Prevalence Estimates by Sex [ Source: ADDM ] [ Year: ALL ]") +
  theme(title = element_text(face = 'bold.italic', color = "darkslategrey"),
        axis.title = element_text(face = 'plain', color = "darkslategrey"),
        legend.position = 'none') +
  coord_flip() + # Rotate chart
  facet_grid(facets = Year ~ .)
```

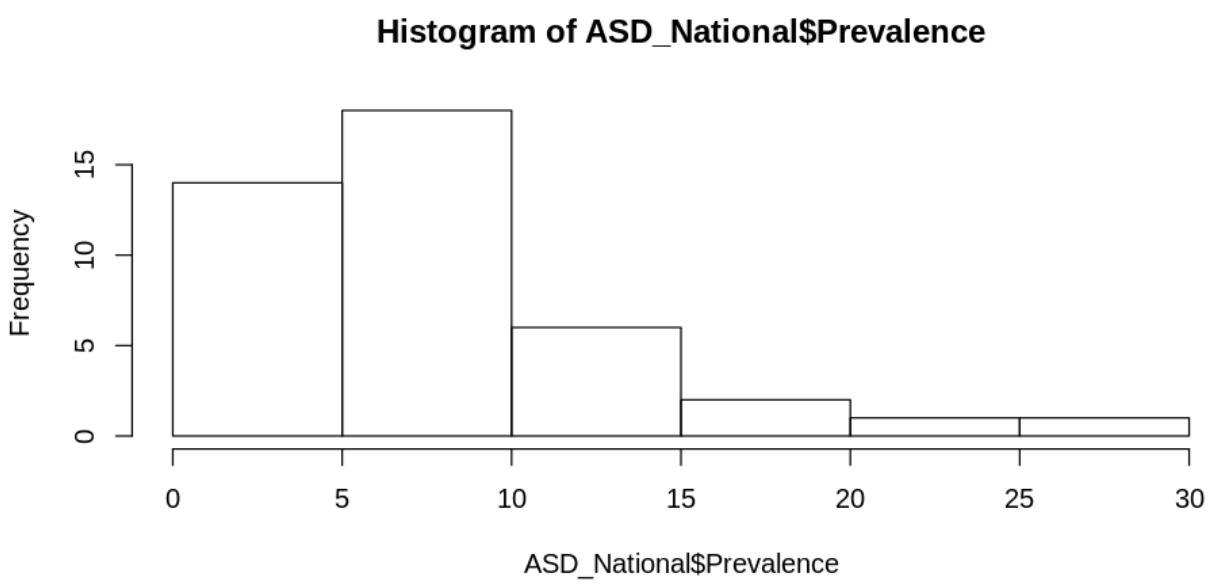
```
Warning message:
“Removed 2 rows containing missing values (position_stack).”Warning message:
“Removed 2 rows containing missing values (geom_text).”
```



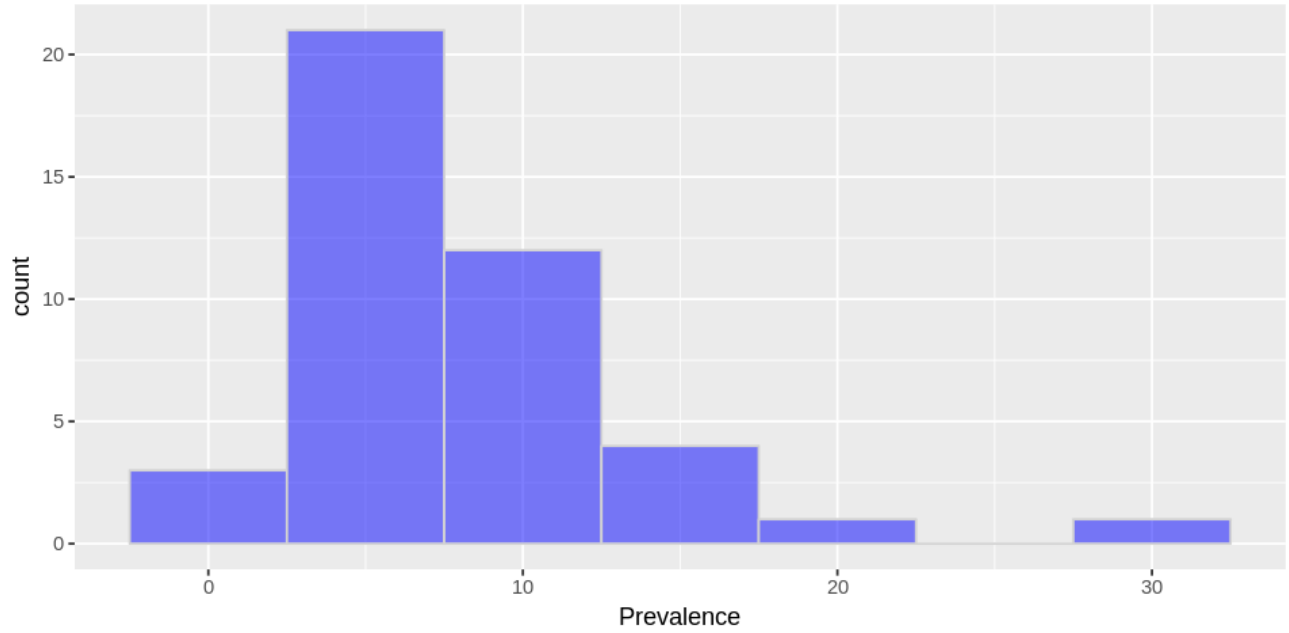
Data Visualisation (Enhanced) - Histogram (distribution of binned continuous variable)

In [159]:
```
# Adjust in-line plot size to M x N
options(repr.plot.width=8, repr.plot.height=4)
```

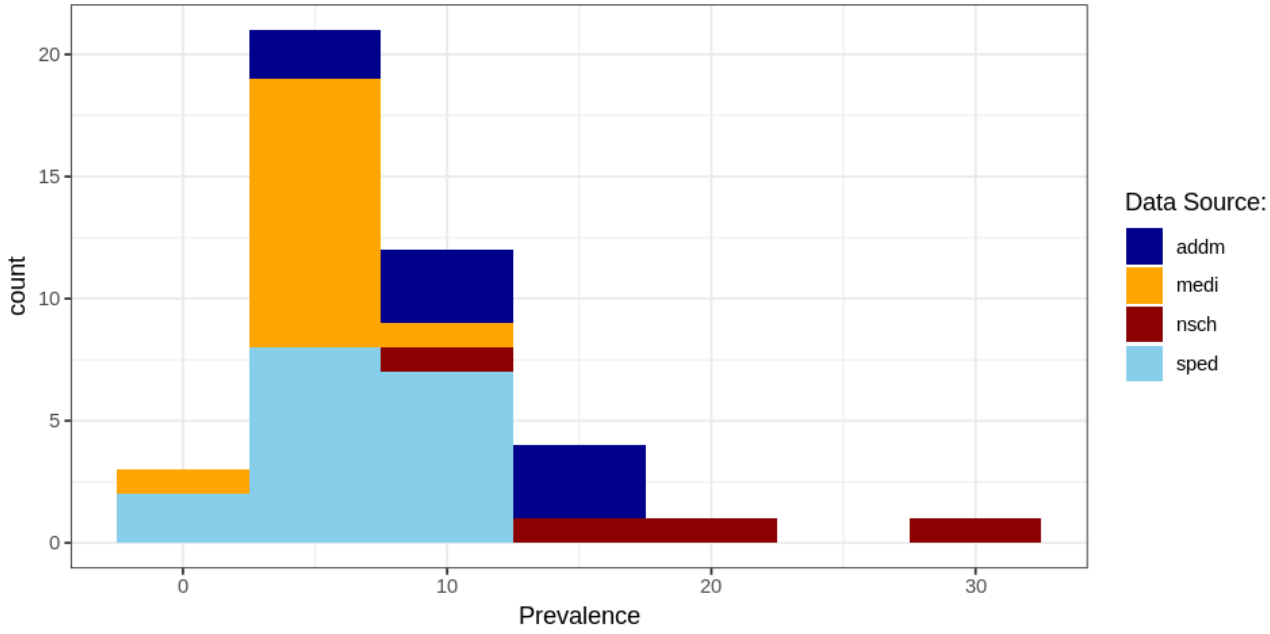
In [160]:
```
# Create histogram using R graphics
hist(ASD_National$Prevalence)
```



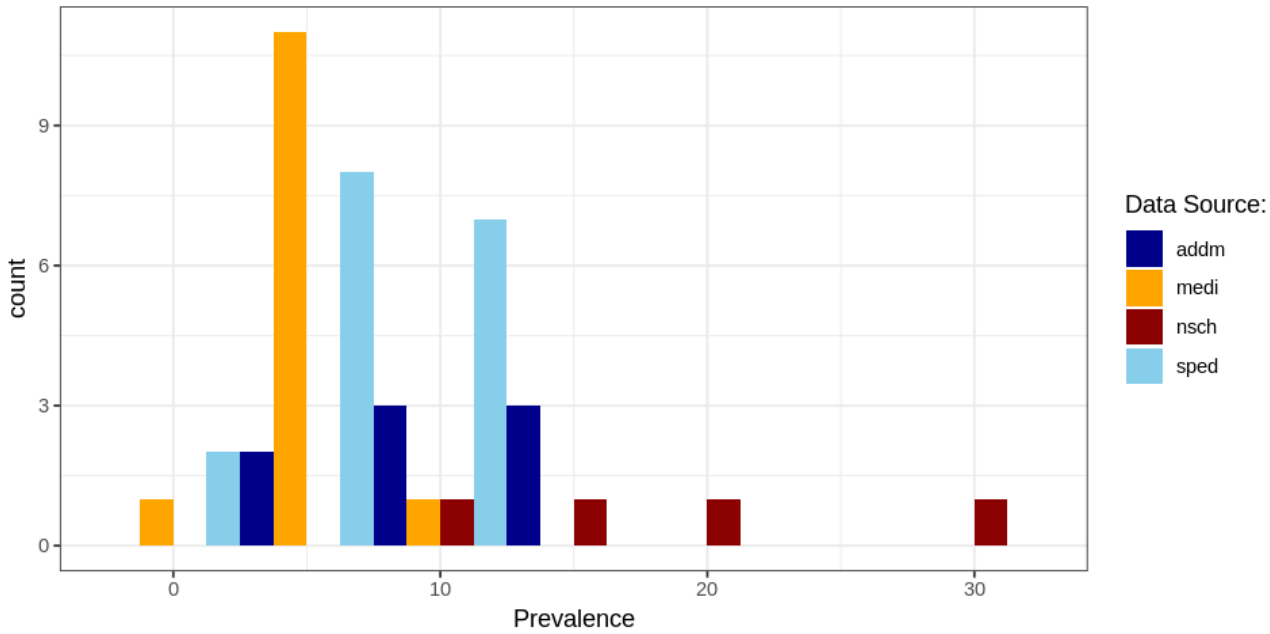
In [161]:
```
# Create histogram using ggplot2
ggplot(ASD_National, aes(x=Prevalence)) +
  geom_histogram(binwidth = 5, fill = "blue", color = "lightgrey", alpha=0.5)
```



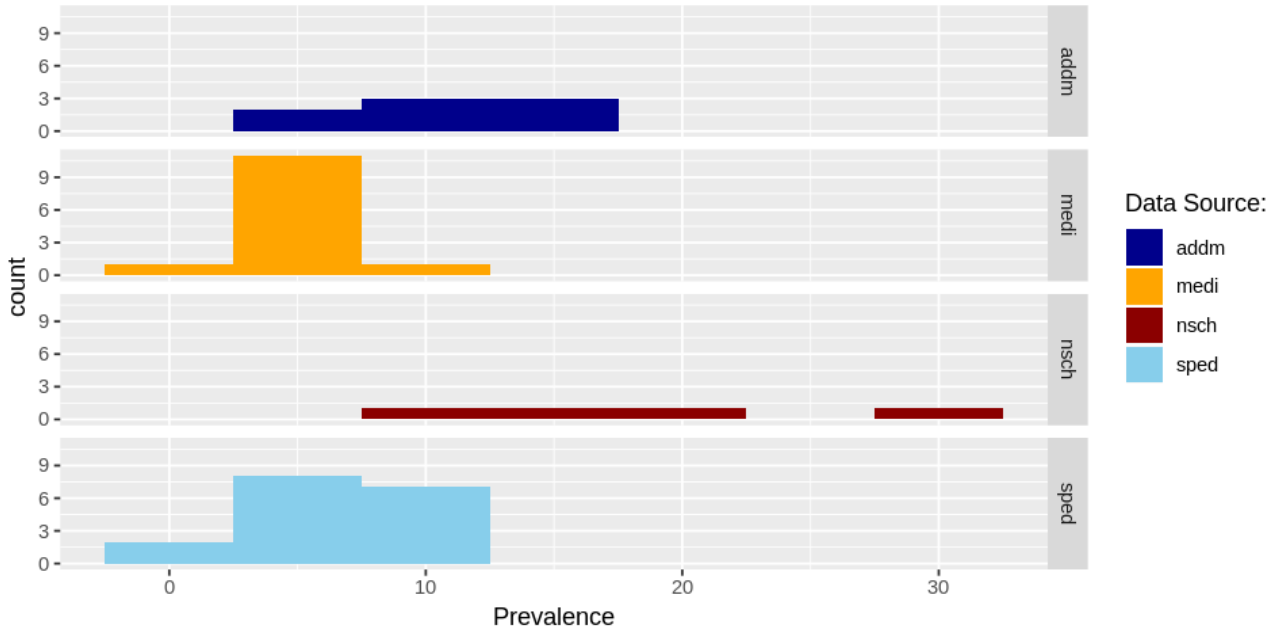
In [162]:
```
# Use color to differentiate sub-group data (Data Source)
ggplot(ASD_National, aes(x=Prevalence, fill = Source)) +
  geom_histogram(binwidth = 5) +
  theme_bw() + theme(legend.position="right") +
  scale_fill_manual("Data Source:", values = c("addm" = "darkblue",
                                               "medi" = "orange",
                                               "nsch" = "darkred",
                                               "sped" = "skyblue"))
```



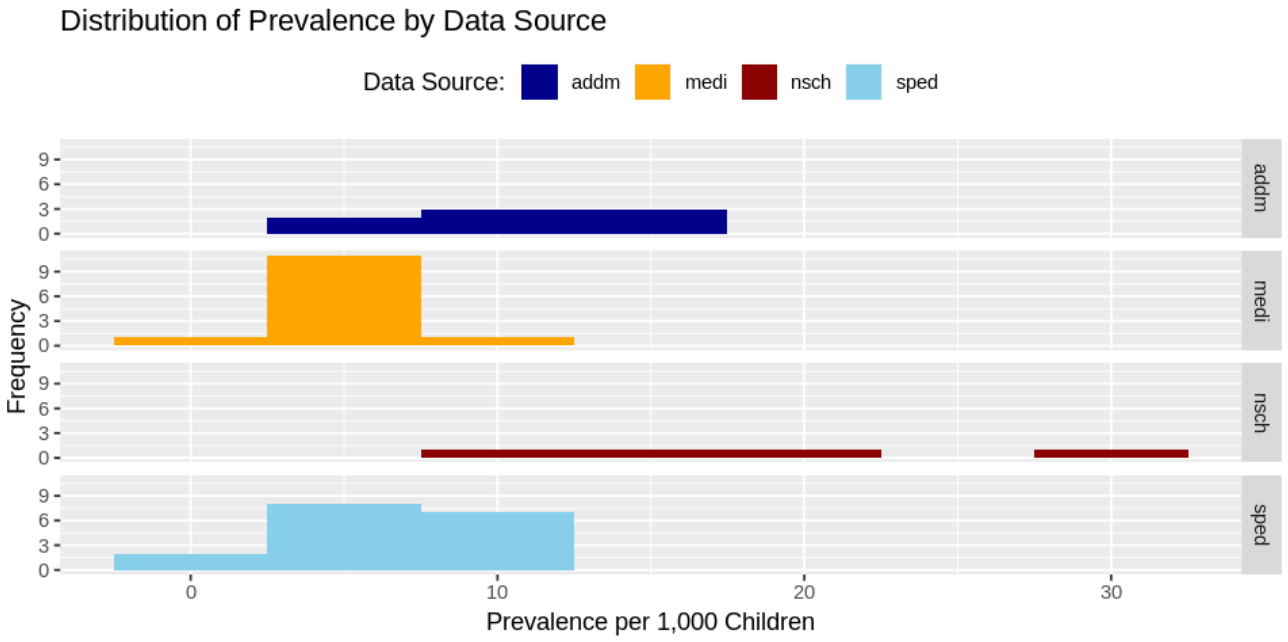
In [163]:
```
# Plot sub-group data side by side, using position="dodge"
ggplot(ASD_National, aes(x=Prevalence, fill = Source)) +
  geom_histogram(binwidth = 5, position="dodge") +
  theme_bw() + theme(legend.position="right") +
  scale_fill_manual("Data Source:", values = c("addm" = "darkblue",
                                               "medi" = "orange",
                                               "nsch" = "darkred",
                                               "sped" = "skyblue"))
```



In [164]:
```
# Split plots using facet_grid()
ggplot(ASD_National, aes(x=Prevalence, fill = Source)) +
  geom_histogram(binwidth = 5) +
  theme(legend.position="right") +
  scale_fill_manual("Data Source:", values = c("addm" = "darkblue",
                                               "medi" = "orange",
                                               "nsch" = "darkred",
                                               "sped" = "skyblue")) +
  facet_grid(facets = Source ~ .)
```



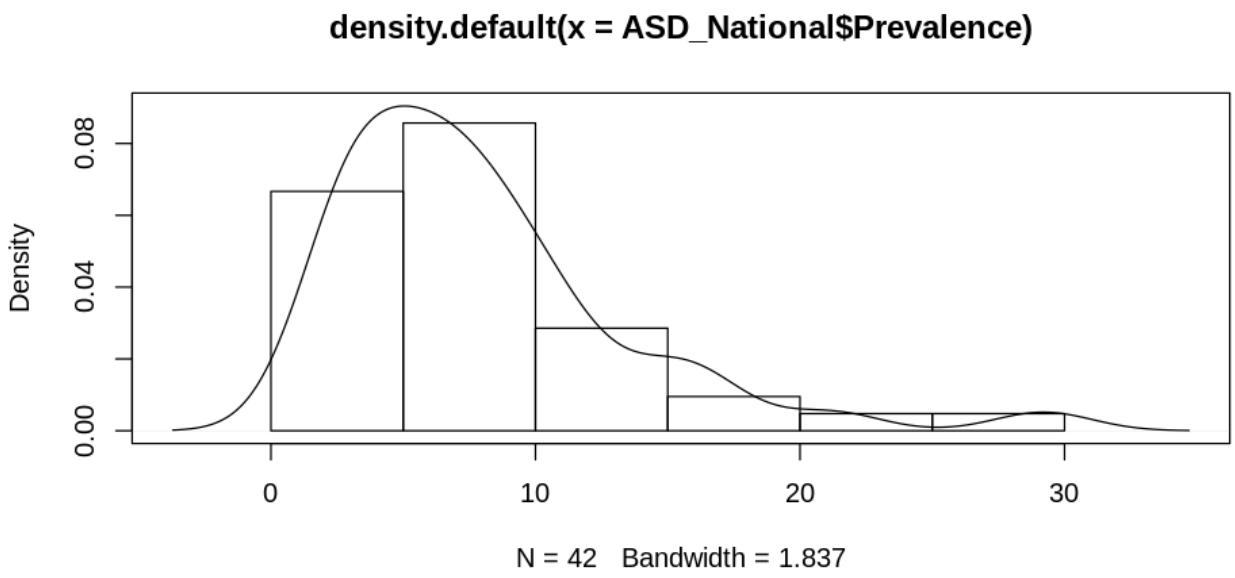
In [165]:
```
# Add title and caption using ggplot2
ggplot(ASD_National, aes(x=Prevalence, fill = Source)) +
  geom_histogram(binwidth = 5) +
  theme(legend.position="top") +
  scale_fill_manual("Data Source:", values = c("addm" = "darkblue",
                                               "medi" = "orange",
                                               "nsch" = "darkred",
                                               "sped" = "skyblue")) +
  labs(x="Prevalence per 1,000 Children",
       y="Frequency",
       title="Distribution of Prevalence by Data Source") +
  facet_grid(facets = Source ~ .)
```



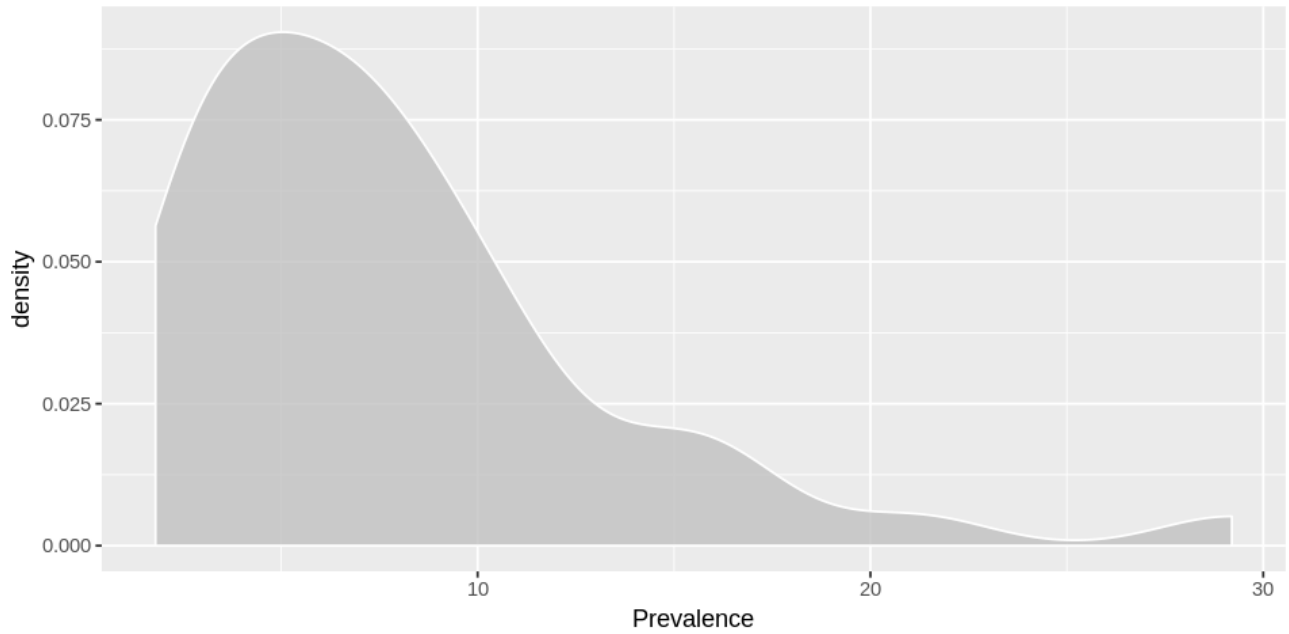
Data Visualisation (Enhanced) - Density plot (distribution for continuous variable normalized to 100% area under curve)

In [166]:
```
# Adjust in-line plot size to M x N
# options(repr.plot.width=8, repr.plot.height=4)
```

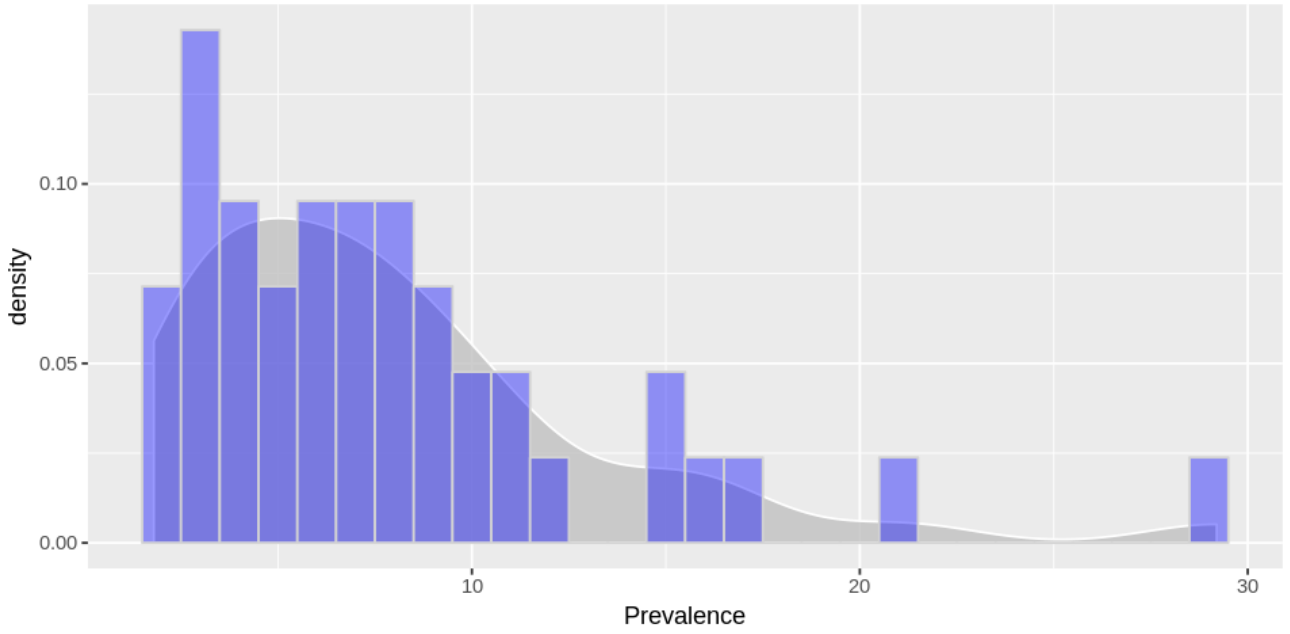
In [167]:
```
# Create plot using R graphics
plot(density(ASD_National$Prevalence))
# Optionally, overlay histogram
hist(ASD_National$Prevalence, probability = TRUE, add = TRUE)
```



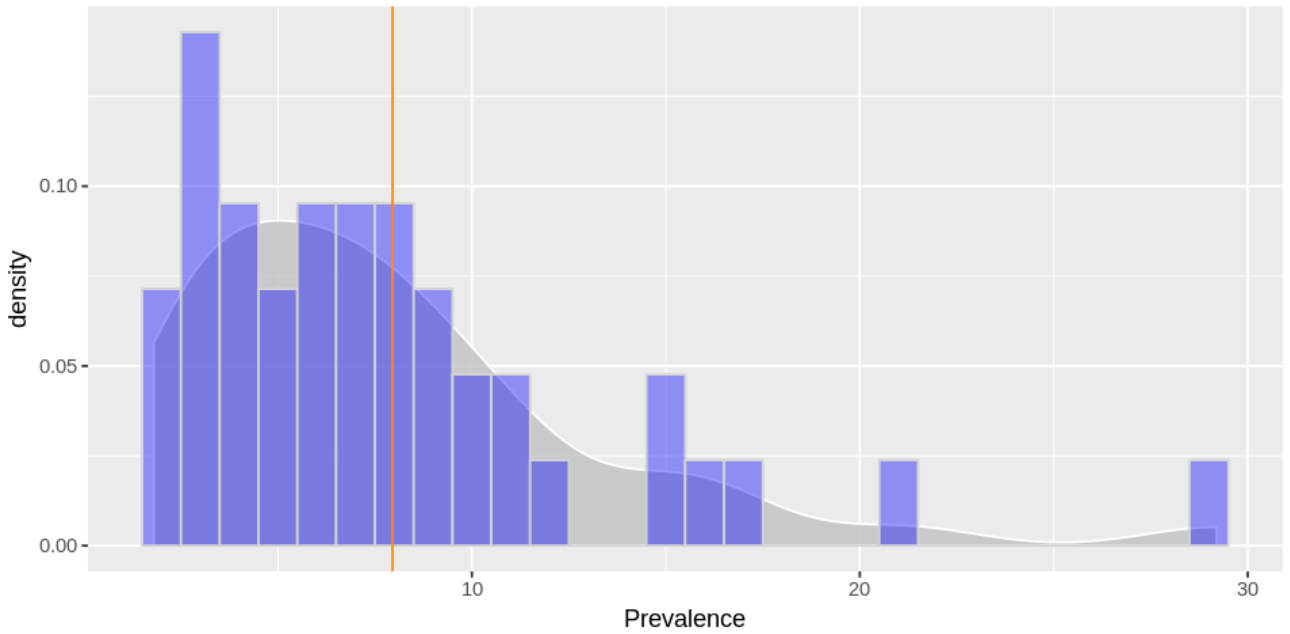
In [168]:
```
# Create plot using ggplot2
p <- ggplot(ASD_National) +
  geom_density(aes(x=Prevalence), fill = "grey", color = "white", alpha=0.75)
p # Show
```



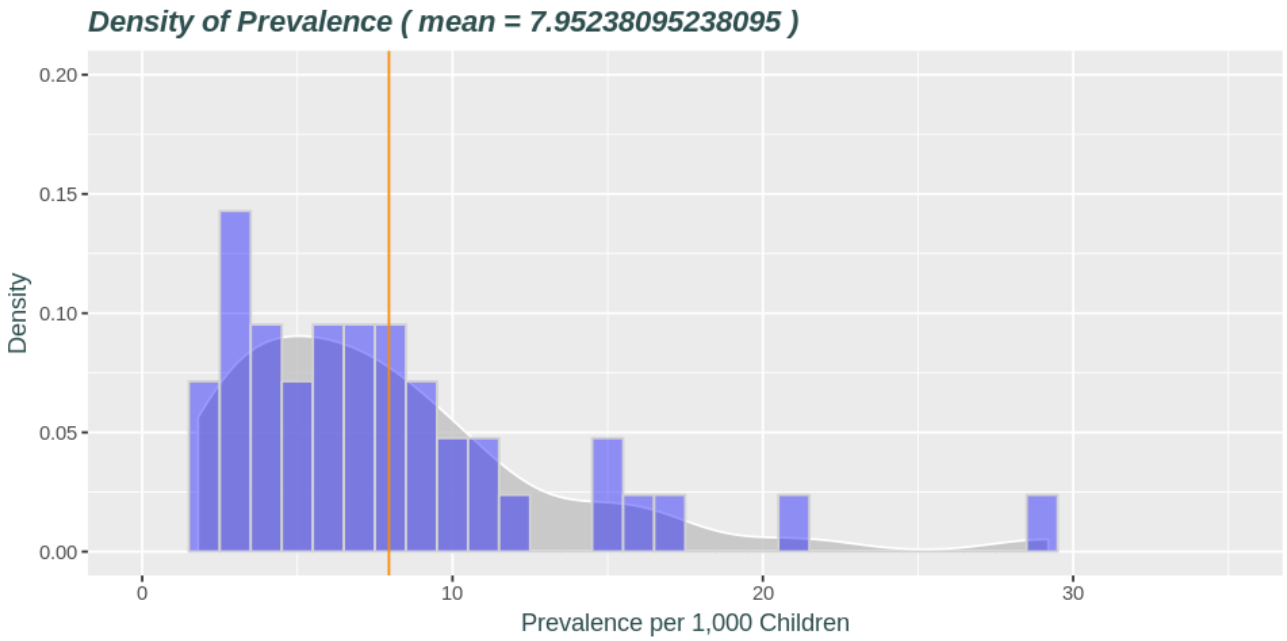
In [169]:
```
# Optionally, overlay histogram
p <- p + geom_histogram(aes(x = Prevalence, y = ..density..), binwidth = 1, fi
p # Show
```



In [170]:
```
# Optionally, overlay Prevalence mean
p <- p + geom_vline(aes(xintercept = mean(ASD_National$Prevalence)), colour="d
p # Show
```

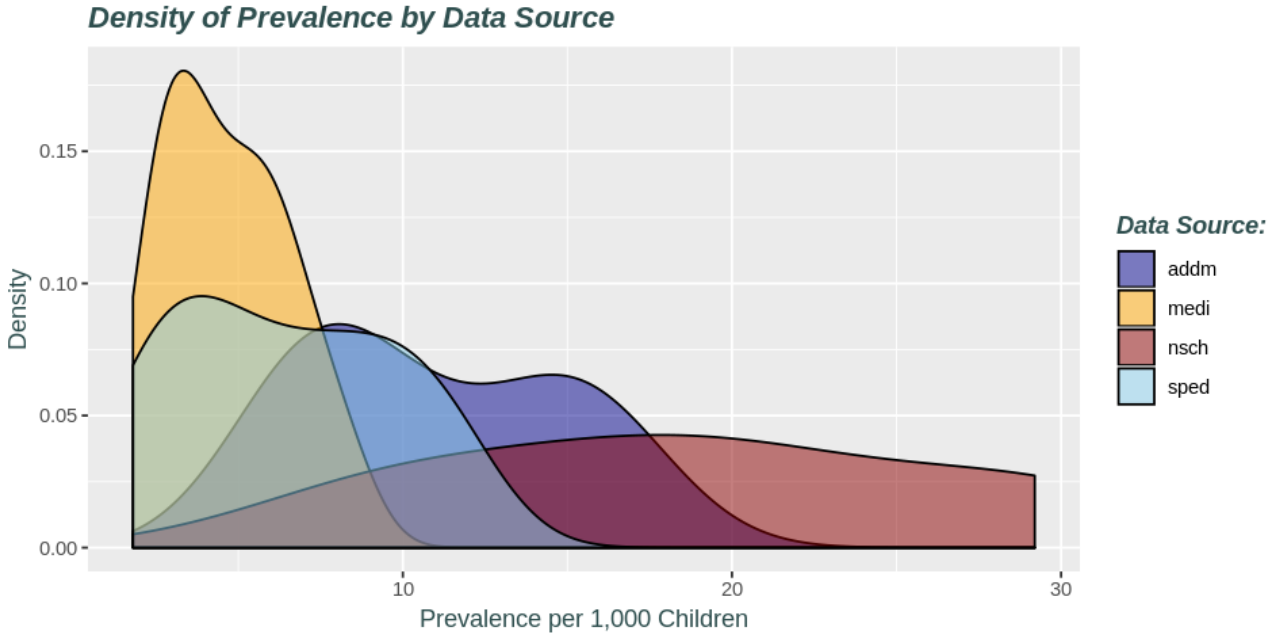


In [171]:
```
# Lastly, add other captions
p <- p + coord_cartesian(xlim=c(0, 35), ylim=c(0, 0.2)) +
  labs(x="Prevalence per 1,000 Children", y="Density",
       title=paste("Density of Prevalence ( mean =", mean(ASD_National$Prevale
  theme(title = element_text(face = 'bold.italic', color = "darkslategrey"),
        axis.title = element_text(face = 'plain', color = "darkslategrey"))
p # Show
```



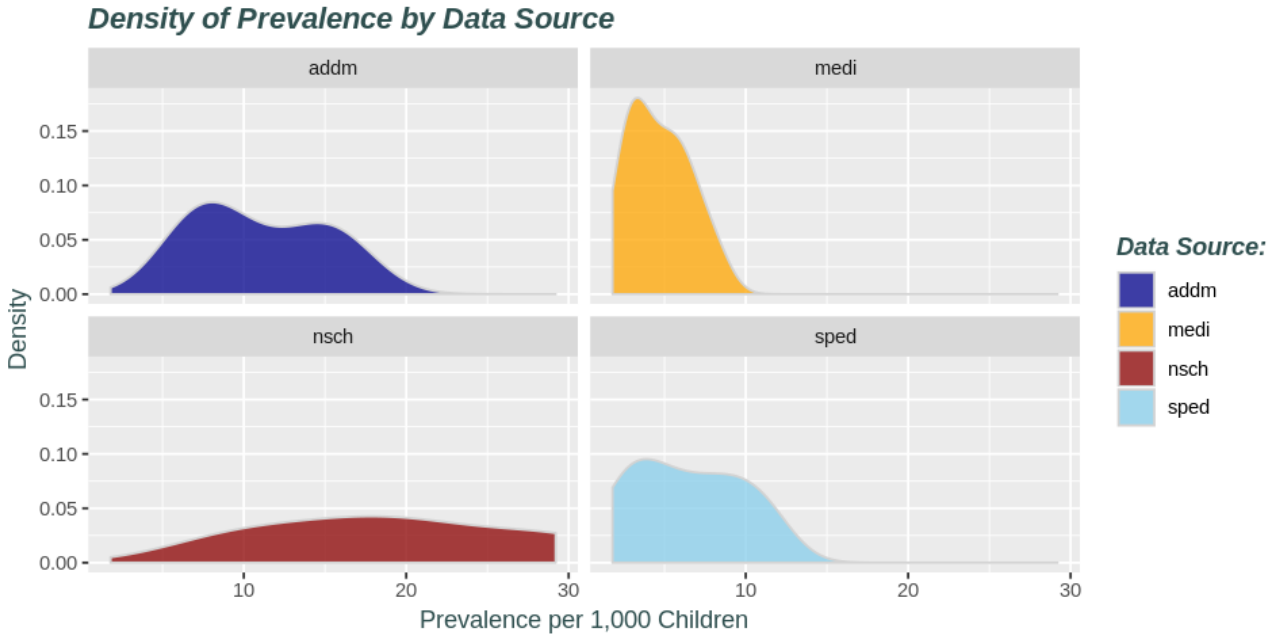
< Prevelance distribution by Data Source >

In [172]:
```
# Prevelance distribution by Data Source
ggplot(ASD_National) + geom_density(aes(x = Prevalence, fill = Source), alpha
  scale_fill_manual("Data Source:", values = c("addm" = "darkblue",
                                               "medi" = "orange",
                                               "nsch" = "darkred",
                                               "sped" = "skyblue")) +
  labs(x="Prevalence per 1,000 Children",
       y="Density",
       title="Density of Prevalence by Data Source") +
  theme(title = element_text(face = 'bold.italic', color = "darkslategrey"),
        axis.title = element_text(face = 'plain', color = "darkslategrey"))
```



< Prevelance distribution by Data Source with split >

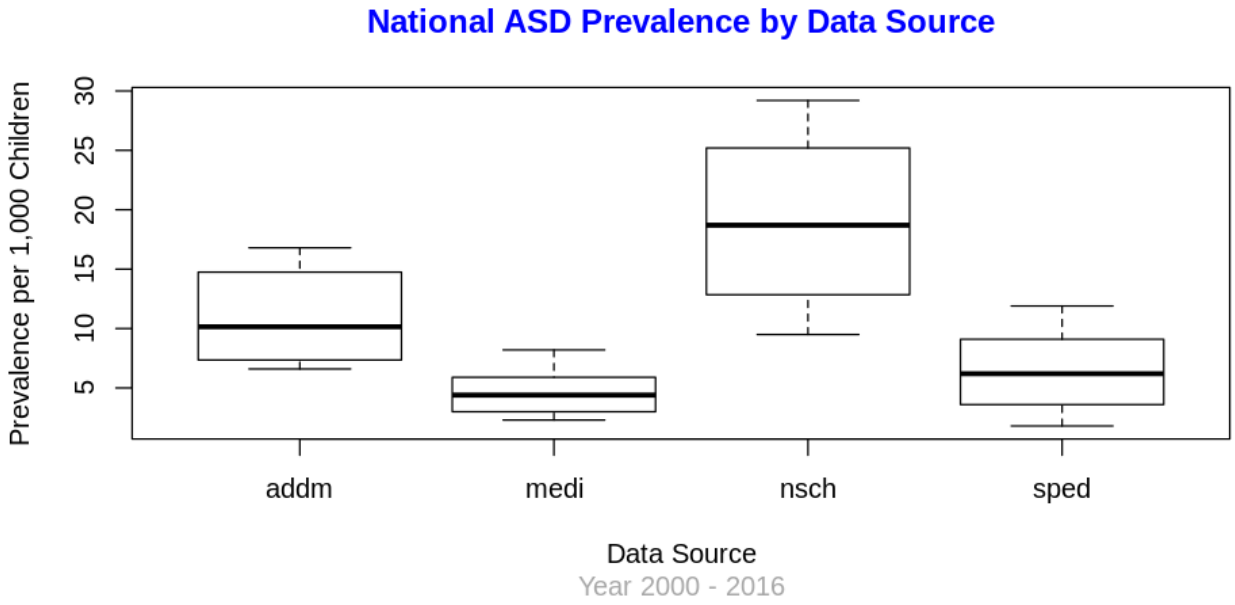
In [173]:
```
# Prevelance distribution by Data Source with split
ggplot(ASD_National) + geom_density(aes(x = Prevalence, fill = Source), colour
  scale_fill_manual("Data Source:", values = c("addm" = "darkblue",
                                               "medi" = "orange",
                                               "nsch" = "darkred",
                                               "sped" = "skyblue")) +
  labs(x="Prevalence per 1,000 Children",
       y="Density",
       title="Density of Prevalence by Data Source") +
  theme(title = element_text(face = 'bold.italic', color = "darkslategrey"),
        axis.title = element_text(face = 'plain', color = "darkslategrey")) +
  facet_wrap(~Source)
```



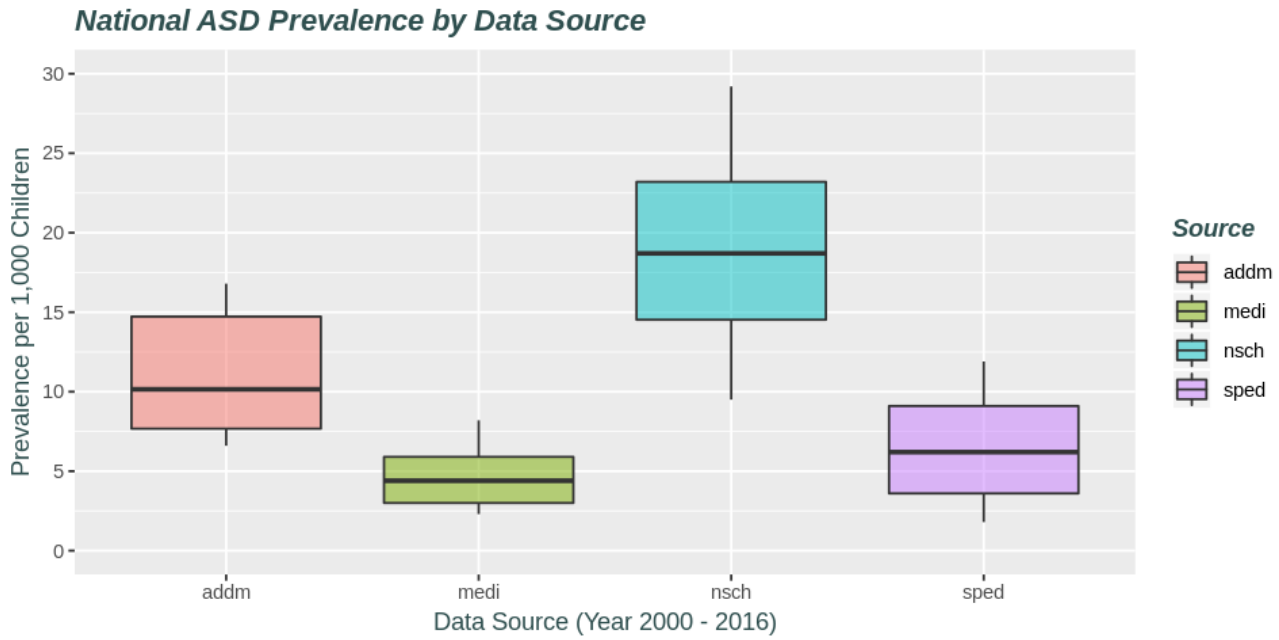
Data Visualisation (Enhanced) - Box plot

In [174]:
```
# Adjust in-line plot size to M x N
options(repr.plot.width=8, repr.plot.height=4)
```

In [175]:
```
# Create plot using R graphics
# Create 'Prevalence' box plots break by 'Source'
boxplot(ASD_National$Prevalence ~ ASD_National$Source,
        main = "National ASD Prevalence by Data Source",
        xlab = "Data Source",
        ylab = "Prevalence per 1,000 Children",
        sub  = "Year 2000 - 2016",
        col.main="blue", col.lab="black", col.sub="darkgrey")
```



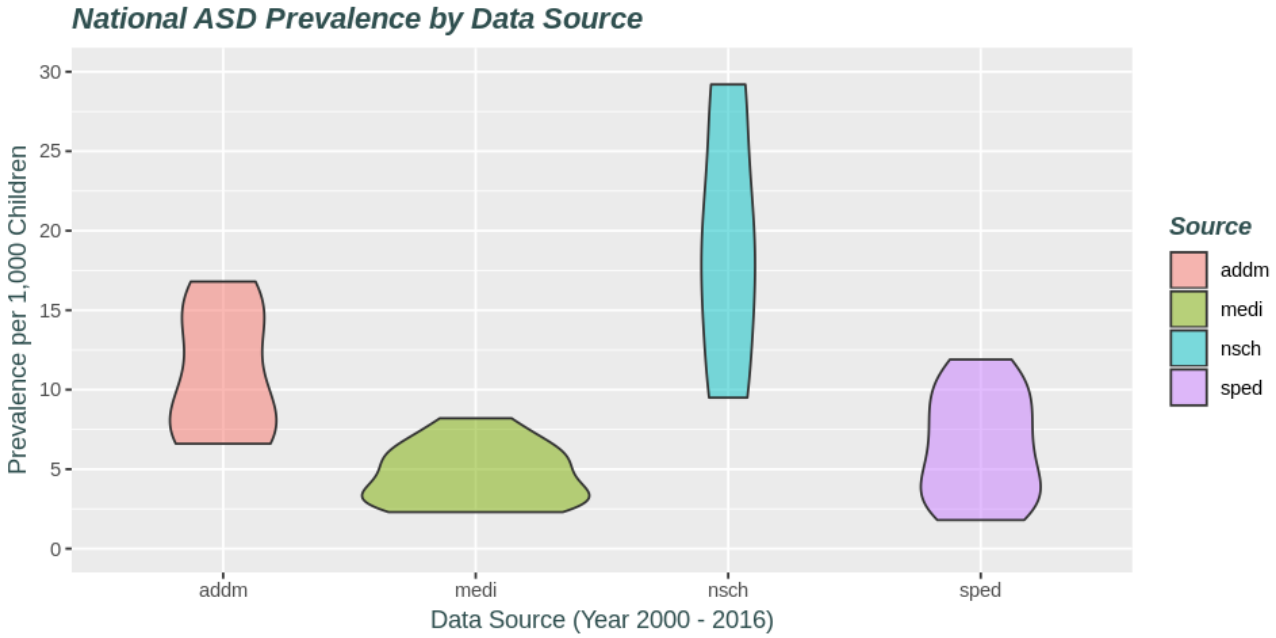
In [176]:
```
# Create box plot using ggplot2
ggplot(ASD_National, aes(x = Source, y = Prevalence, fill = Source)) +
  geom_boxplot(alpha = 0.5) +
  scale_y_continuous(name = "Prevalence per 1,000 Children",
                     breaks = seq(0, 30, 5),
                     limits=c(0, 30)) +
  scale_x_discrete(name = "Data Source (Year 2000 - 2016)") +
  ggtitle("National ASD Prevalence by Data Source") +
  theme(title = element_text(face = 'bold.italic', color = "darkslategrey"),
        axis.title = element_text(face = 'plain', color = "darkslategrey"))
```



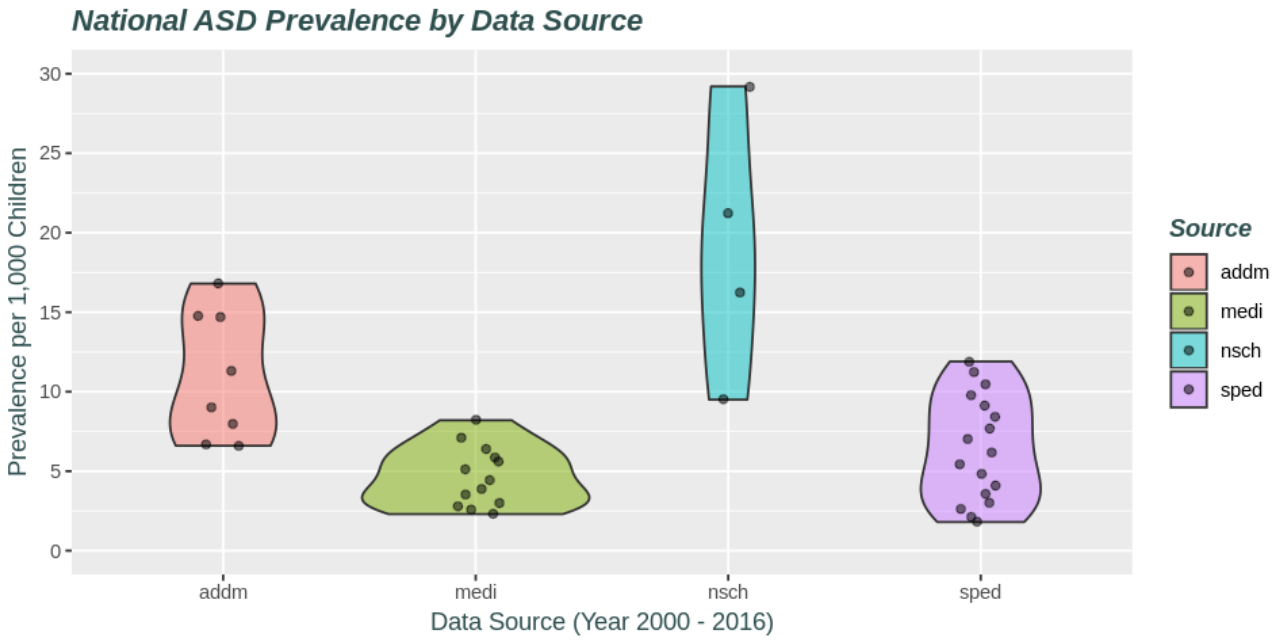
Data Visualisation (Enhanced) - Violin plot

In [177]:
```
# Adjust in-line plot size to M x N
# options(repr.plot.width=8, repr.plot.height=4)
```

In [178]:
```
# Create plot using ggplot2
ggplot(ASD_National, aes(x = Source, y = Prevalence, fill = Source)) +
  geom_violin(alpha = 0.5) +
  scale_y_continuous(name = "Prevalence per 1,000 Children",
                     breaks = seq(0, 30, 5),
                     limits=c(0, 30)) +
  scale_x_discrete(name = "Data Source (Year 2000 - 2016)") +
  ggtitle("National ASD Prevalence by Data Source") +
  theme(title = element_text(face = 'bold.italic', color = "darkslategrey"),
        axis.title = element_text(face = 'plain', color = "darkslategrey"))
```

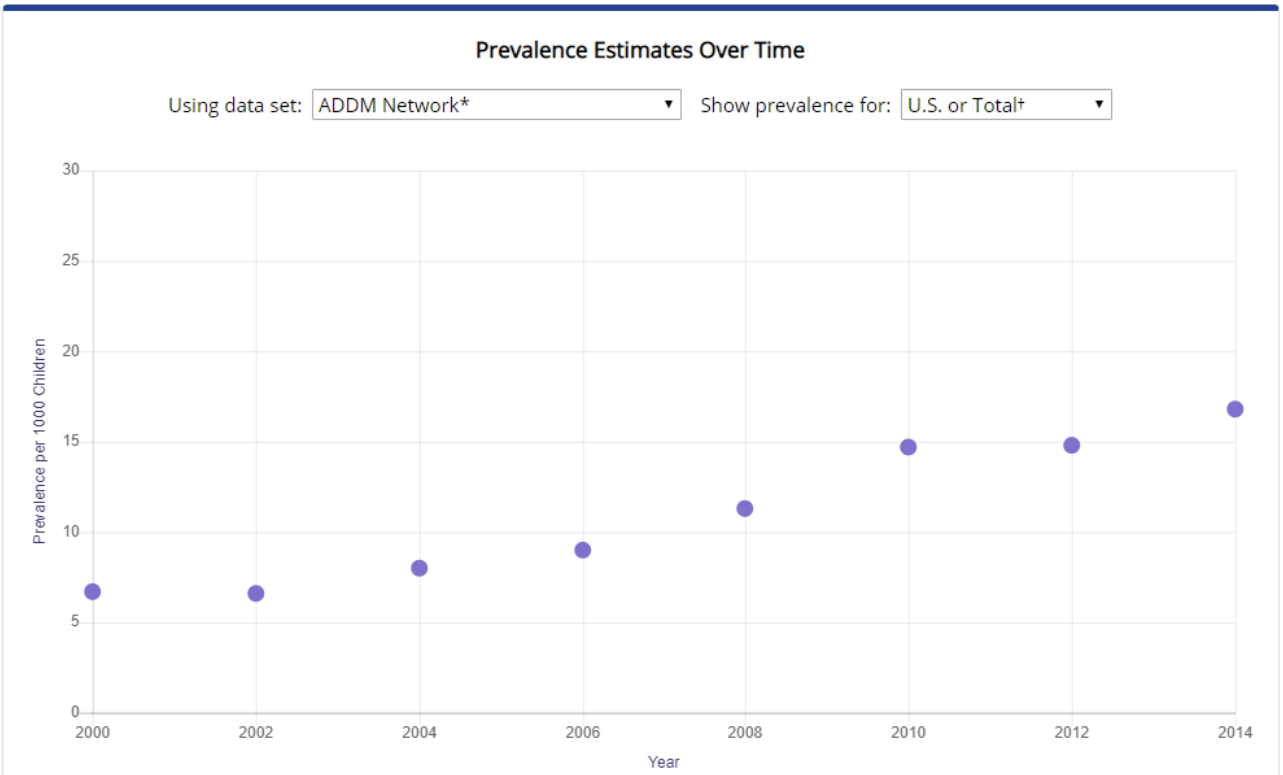


In [179]:
```
# Create plot using ggplot2
ggplot(ASD_National, aes(x = Source, y = Prevalence, fill = Source)) +
  geom_violin(alpha = 0.5) +
  geom_jitter(alpha = 0.5, position = position_jitter(width = 0.1)) + # Overla
#  coord_flip() + # Uncomment to flip x-y axis
  scale_y_continuous(name = "Prevalence per 1,000 Children",
                     breaks = seq(0, 30, 5),
                     limits=c(0, 30)) +
  scale_x_discrete(name = "Data Source (Year 2000 - 2016)") +
  ggtitle("National ASD Prevalence by Data Source") +
  theme(title = element_text(face = 'bold.italic', color = "darkslategrey"),
        axis.title = element_text(face = 'plain', color = "darkslategrey"))
```



Data Visualisation (Enhanced) - Line chart

Data Visualisation (Enhanced) - [CDC] REPORTED PREVALENCE HAS CHANGED OVER TIME

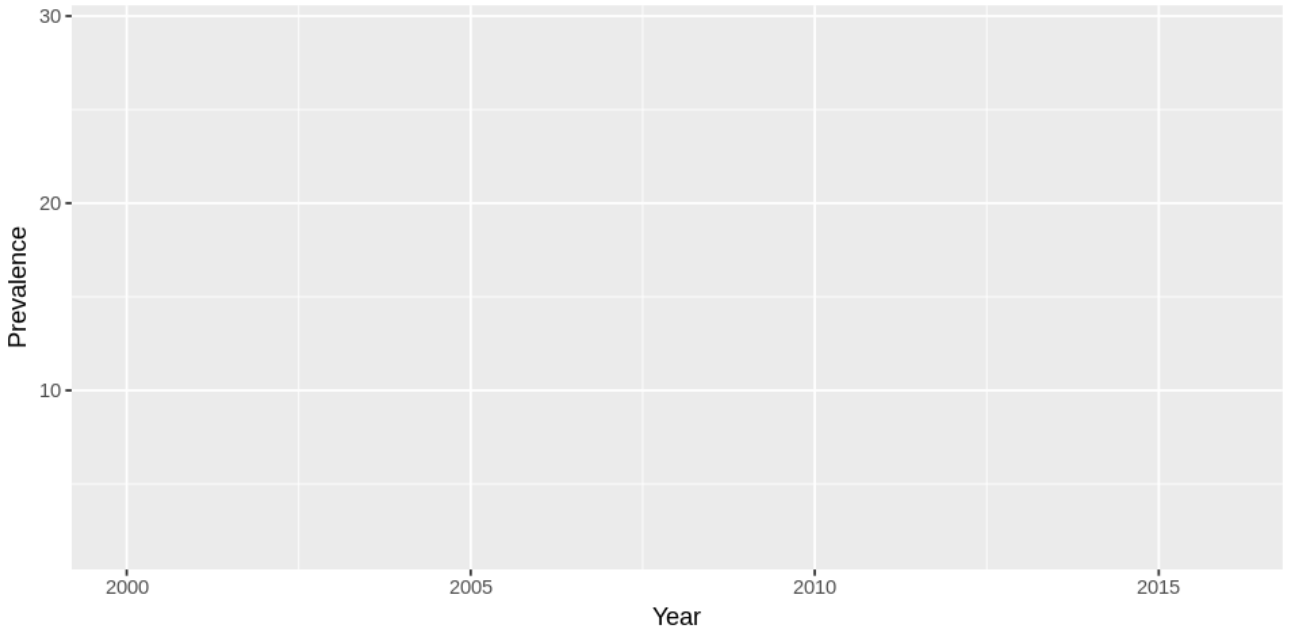


Note: *Hover your mouse over data points above to show prevalence by year.*
**ADDM Network data only represent a selection of sites within states (but not entire states) that were funded during each project cycle; therefore, data are not available for the entire United States.*
***ADDM estimate = the total for all sites combined.*
*†**NSCH data are not comparable over time as data collection methods changed. See technical notes for further details.***
††For NSCH data, data are suppressed when the width of the confidence interval exceeds 1.2 times the point estimate. This is the same approach that is recommended by NSCH here and our point estimates (and suppressed data points) match those on childhealthdata.org for 2016-2017.
†††If NSCH 2016 data are selected, combined 2016-2017 estimates are shown.

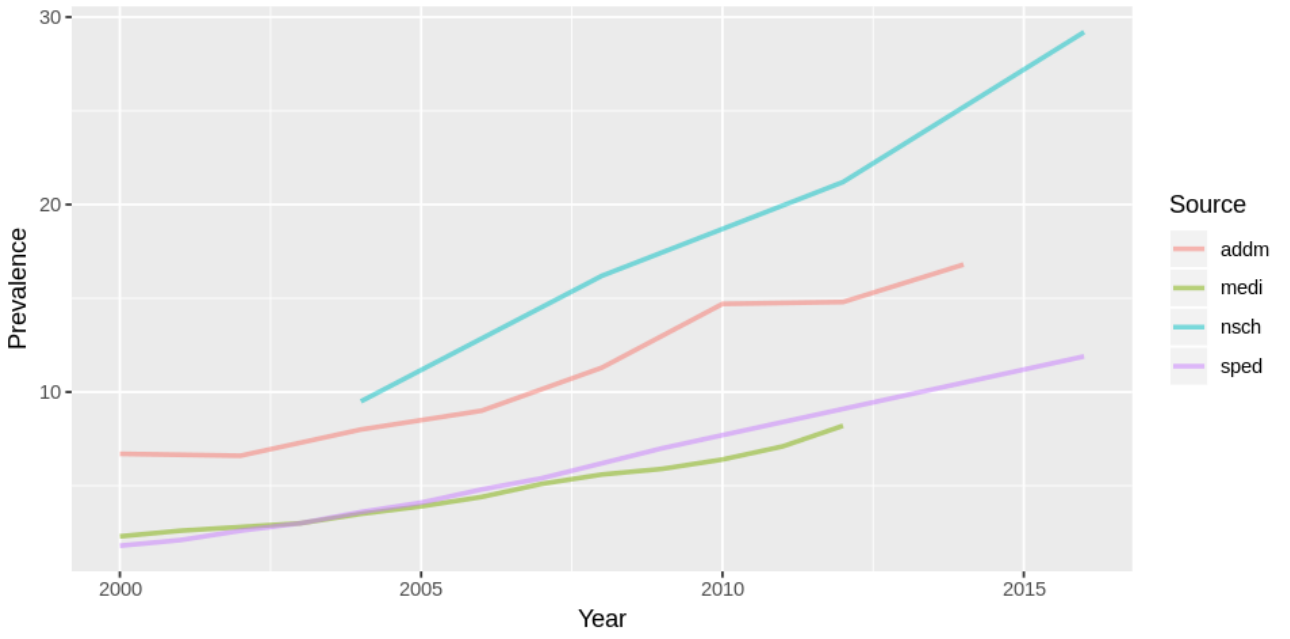
Data Visualisation (Enhanced) - [R] REPORTED PREVALENCE HAS CHANGED OVER TIME [Source: ALL]

In [180]:
```
# Adjust in-line plot size to M x N
options(repr.plot.width=8, repr.plot.height=4)
```

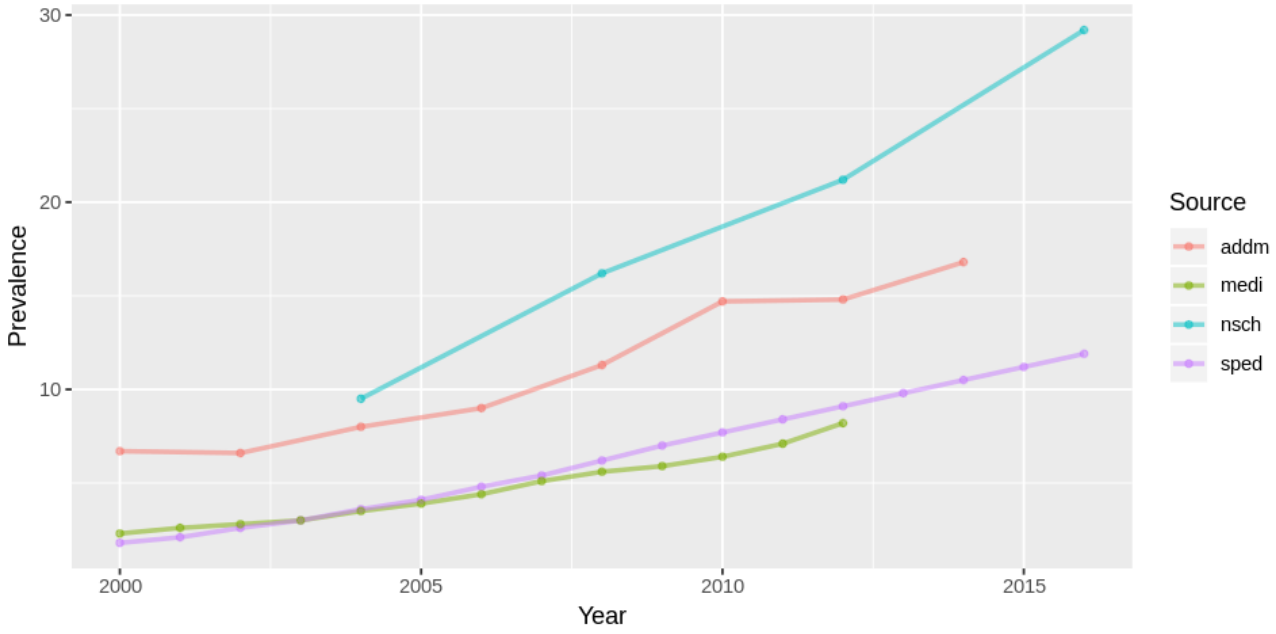
In [181]:
```
# ---------------------------------
# Build chart/plot layer by layer
# ---------------------------------

# Define a ggplot graphic object; provide data and x y for use
p <- ggplot(ASD_National, aes(x = Year, y = Prevalence))
# Show plot
p
```



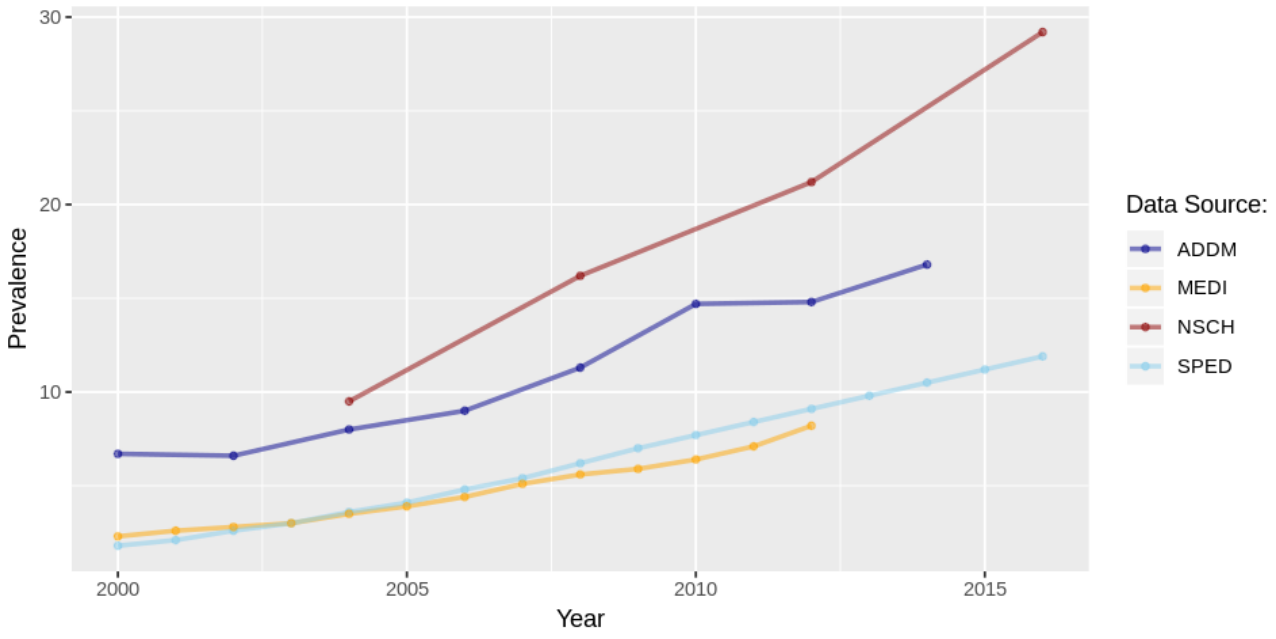
In [182]:
```
# Select (add) line chart type:
p <- p + geom_line(aes(color = Source),
                   linetype = "solid",  # http://sape.inf.usi.ch/quick-referen
                   size=1,
                   alpha=0.5)
# Show plot
p
```



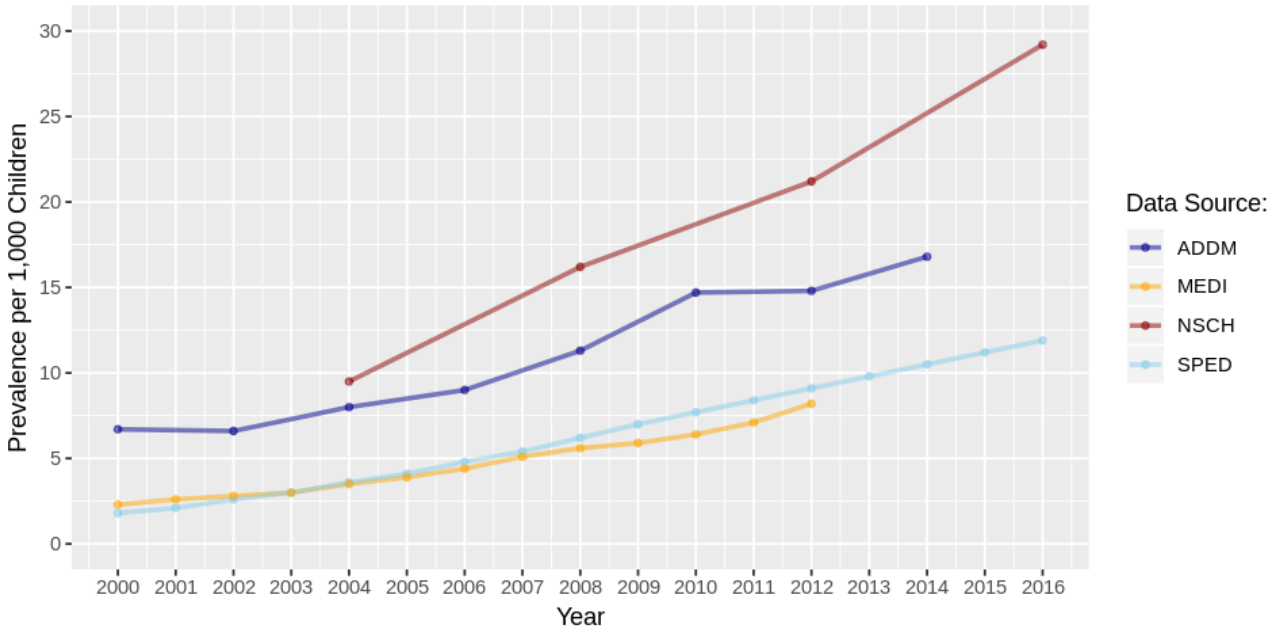
In [183]:
```
# Select (add) points to chart:
p <- p + geom_point(aes(color = Source),
                    size=2,
                    shape=20,
                    alpha=0.5)
# Show plot
p
```



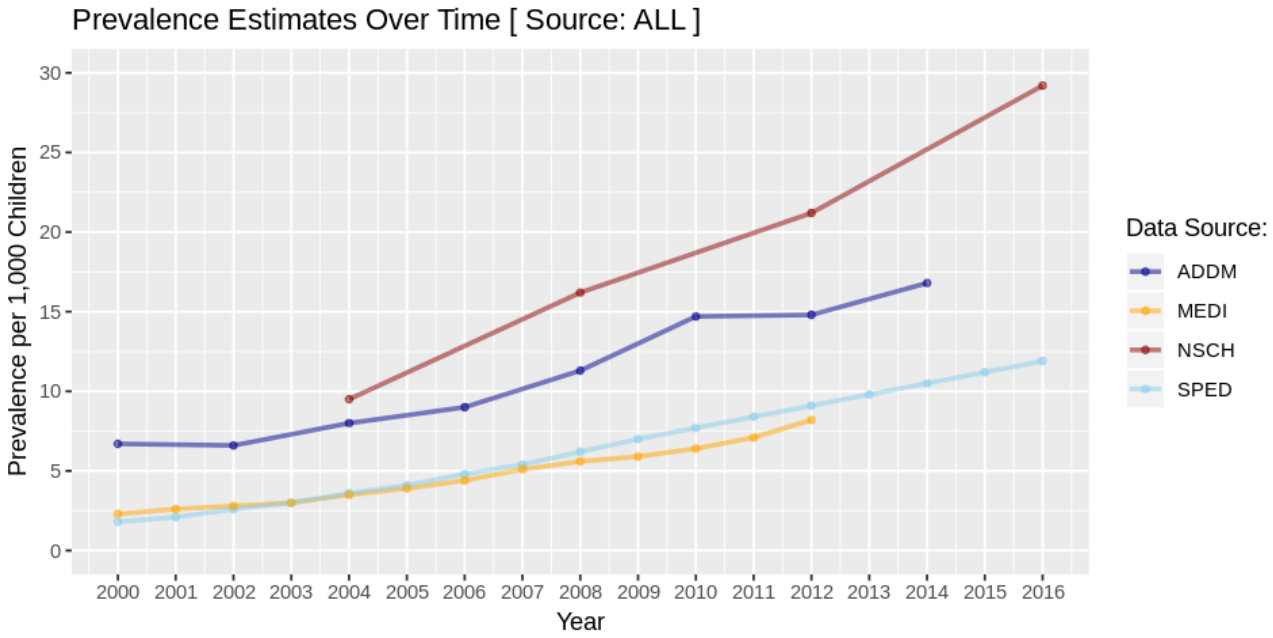
In [184]:
```
# Customize line color and legend name:
p <- p + scale_color_manual("Data Source:",
                            labels = c('ADDM', 'MEDI', 'NSCH', 'SPED'),
                            values = c("addm" = "darkblue",
                                       "medi" = "orange",
                                       "nsch" = "darkred",
                                       "sped" = "skyblue"))
# Show plot
p
```



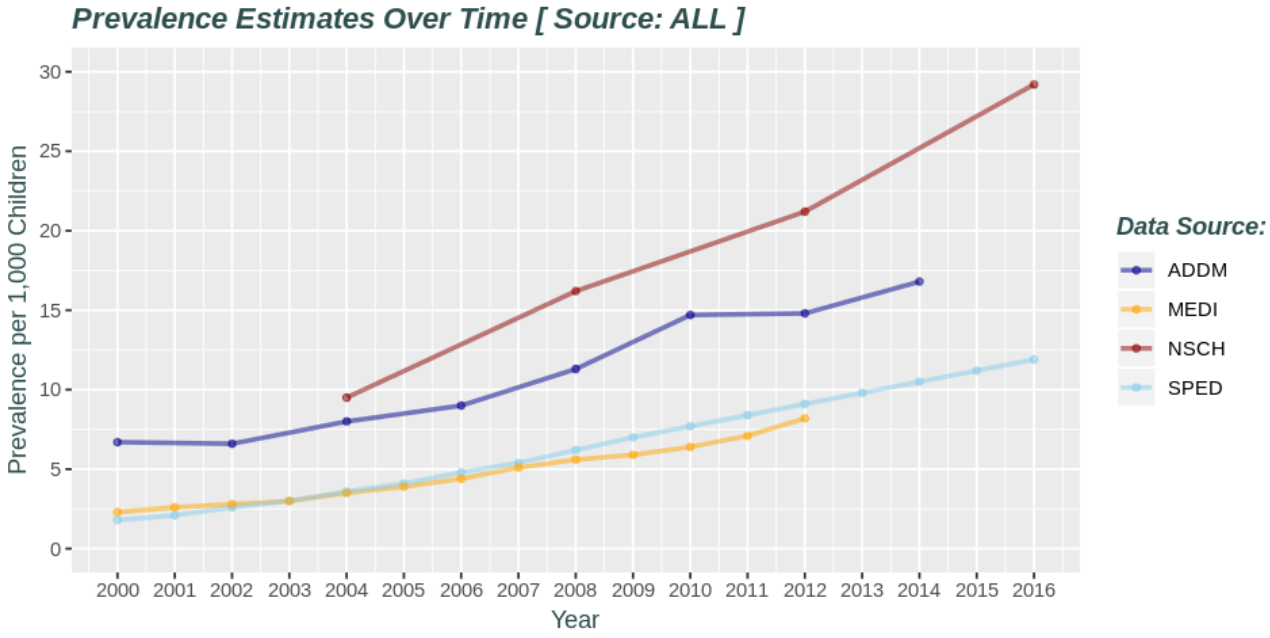
In [185]:
```
# Adjust x and y axis, scale, limit and labels:
p <- p + scale_y_continuous(name = "Prevalence per 1,000 Children",
                            breaks = seq(0, 30, 5),
                            limits=c(0, 30)) +
  scale_x_continuous(name = "Year",
                     breaks = seq(2000, 2016, 1),
                     limits = c(2000, 2016))
# Show plot
p
```



In [186]:
```
# Customise chart title:
p <- p + ggtitle("Prevalence Estimates Over Time [ Source: ALL ]")
# Show plot
p
```

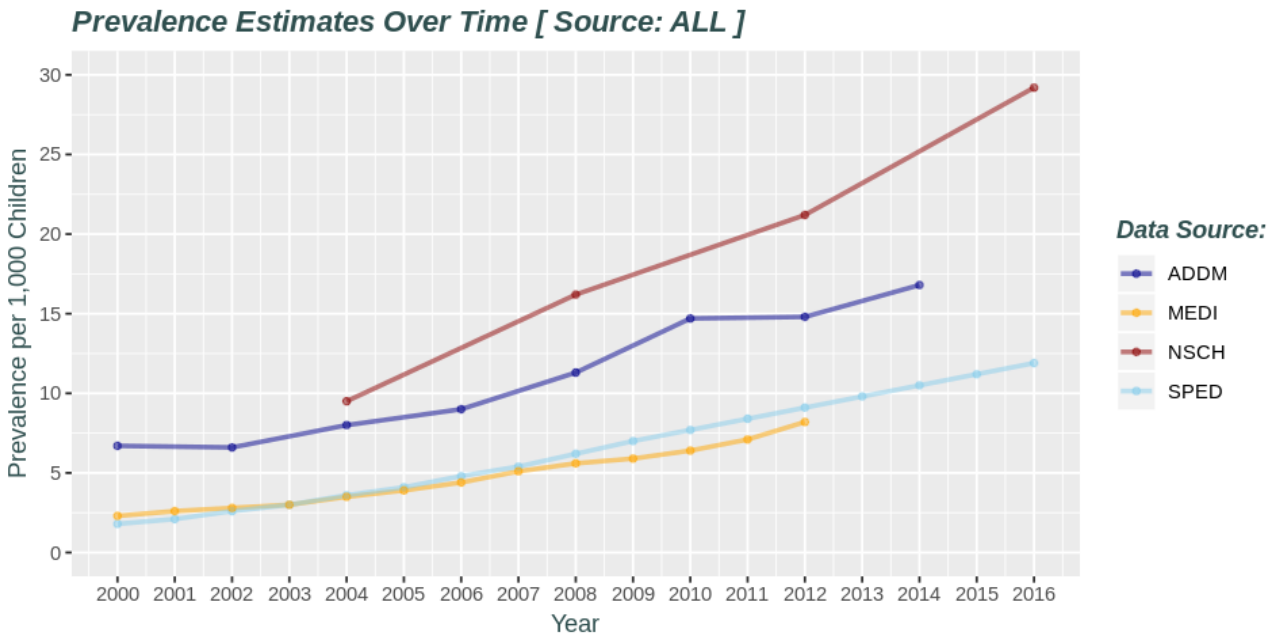


In [187]:
```
# Customise chart title and axis labels:
p <- p + theme(title = element_text(face = 'bold.italic', color = "darkslategr
               axis.title = element_text(face = 'plain', color = "darkslategre
# Show plot
p
```



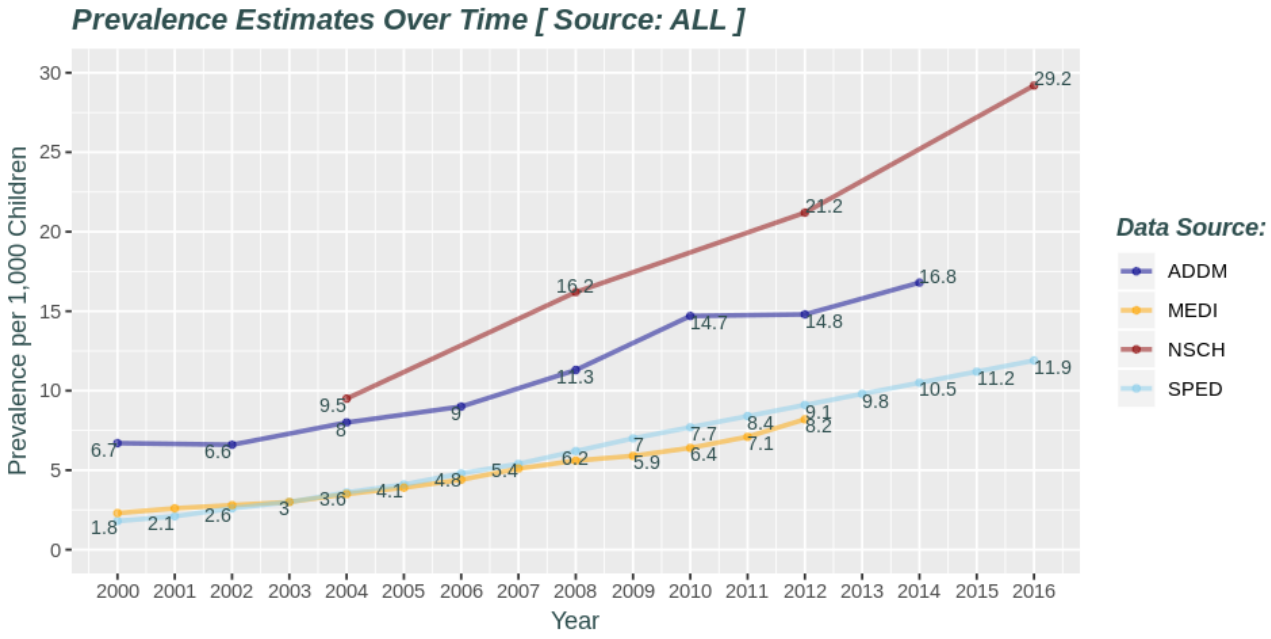
Consolidate above code into one chunk:

In [188]:
```
# ---------------------------------
# Consolidate above code into one chunk
# ---------------------------------
p <- ggplot(ASD_National, aes(x = Year, y = Prevalence)) +
  geom_line(aes(color = Source),
            linetype = "solid",  # http://sape.inf.usi.ch/quick-reference/ggpl
            size=1,
            alpha=0.5) +
  geom_point(aes(color = Source),
             size=2,
             shape=20,
             alpha=0.5) +
  scale_color_manual("Data Source:",
                     labels = c('ADDM', 'MEDI', 'NSCH', 'SPED'),
                     values = c("addm" = "darkblue",
                                "medi" = "orange",
                                "nsch" = "darkred",
                                "sped" = "skyblue")) +
  scale_y_continuous(name = "Prevalence per 1,000 Children",
                     breaks = seq(0, 30, 5),
                     limits=c(0, 30)) +
  scale_x_continuous(name = "Year",
                     breaks = seq(2000, 2016, 1),
                     limits = c(2000, 2016)) +
  ggtitle("Prevalence Estimates Over Time [ Source: ALL ]") +
  theme(title = element_text(face = 'bold.italic', color = "darkslategrey"),
        axis.title = element_text(face = 'plain', color = "darkslategrey"))
# Show plot
p
```



Optionally, display data values/labels:

In [189]:
```
# Optionally, displaydata values/labels
p + geom_text(aes(label = round(Prevalence, 1)), # Values are rounded for disp
              vjust = "outward",
              #          nudge_y = 0.2, # optionally life the text
              hjust = "outward",
              check_overlap = TRUE,
              size = 3, # size of textual data label
              col = 'darkslategrey')
```



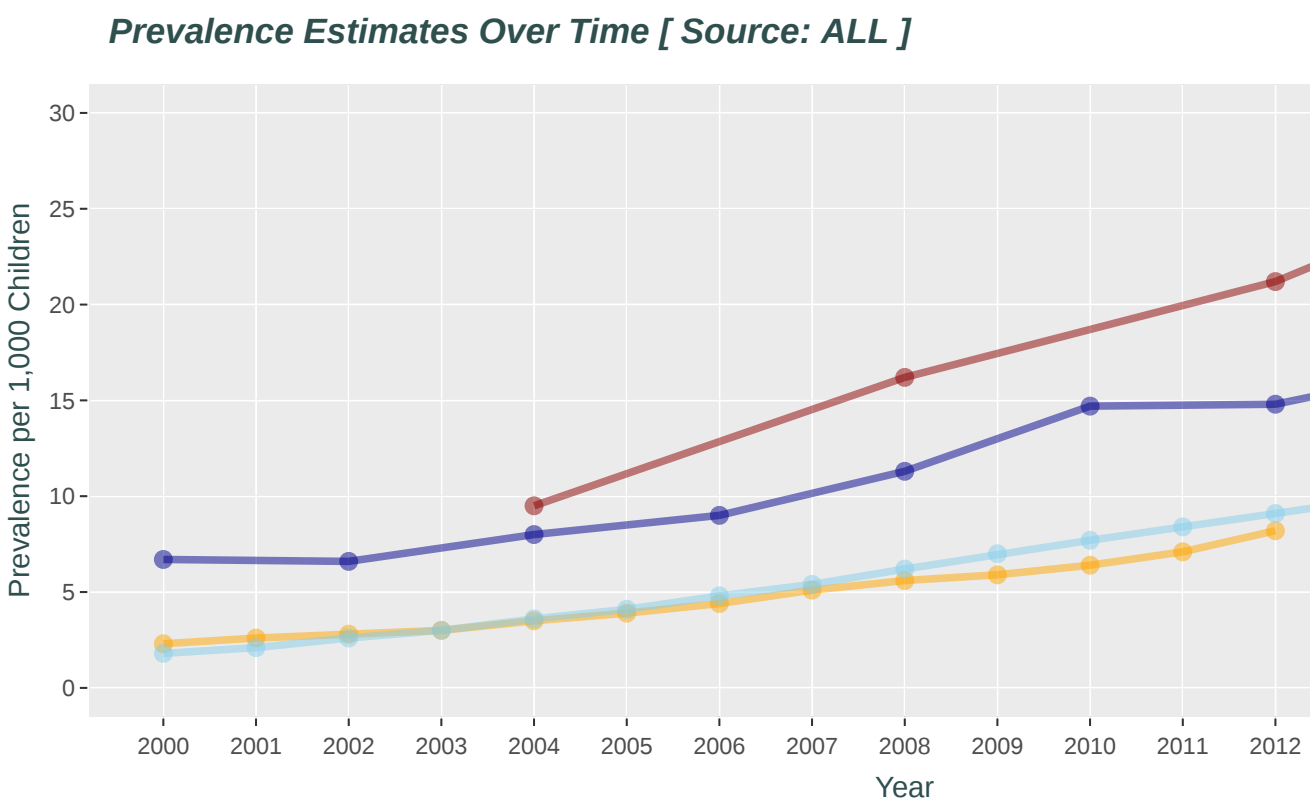
Data Visualisation (Enhanced) - Dynamic Visualisation with plotly

In [190]:
```
if(!require(plotly)){install.packages("plotly")}
library(plotly)
```

```
Loading required package: plotly

Attaching package: ‘plotly’

The following object is masked from ‘package:ggplot2’:

    last_plot

The following object is masked from ‘package:stats’:

    filter

The following object is masked from ‘package:graphics’:

    layout
```

Create ploly graph object from ggplot graph object:

In [191]:
```
p_dynamic <- p
p_dynamic <- ggplotly(p_dynamic)
p_dynamic
```



Data Visualisation (Enhanced) - Use themes as aesthetic template

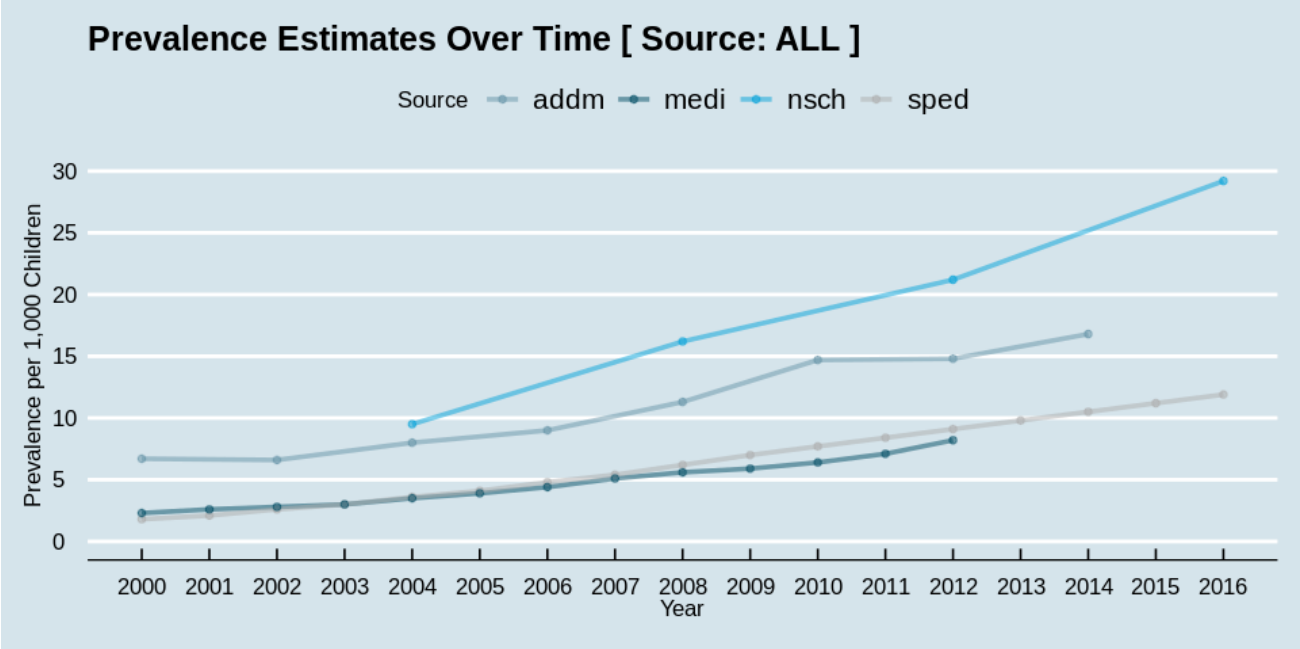
In [192]:
```
if(!require(ggthemes)){install.packages("ggthemes")}
library('ggthemes')
```

```
Loading required package: ggthemes
```

Theme of the Economist magazine:

In [193]:
```
# Theme of the economist magazine:
p + theme_economist() + scale_colour_economist()
```

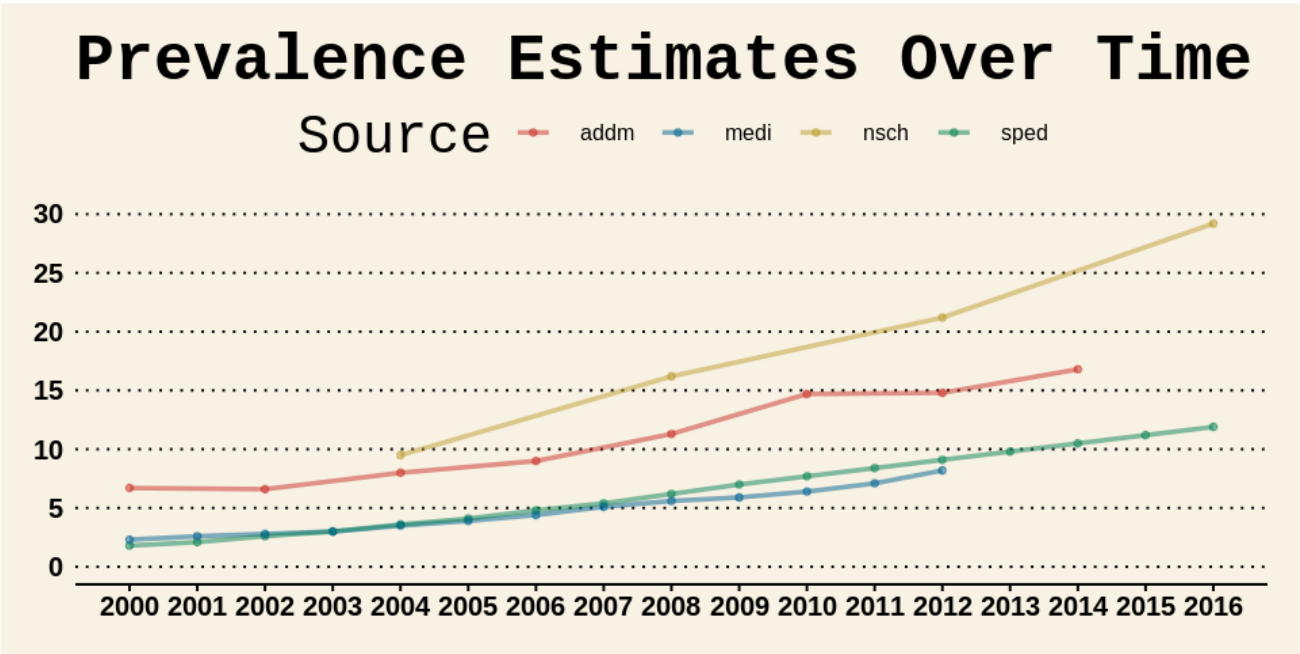
```
Scale for 'colour' is already present. Adding another scale for 'colour',
which will replace the existing scale.
```



Theme of the Wall Street Journal:

In [194]:
```
# Theme of the Wall Street Journal:
p + theme_wsj() + scale_colour_wsj("colors6")
```

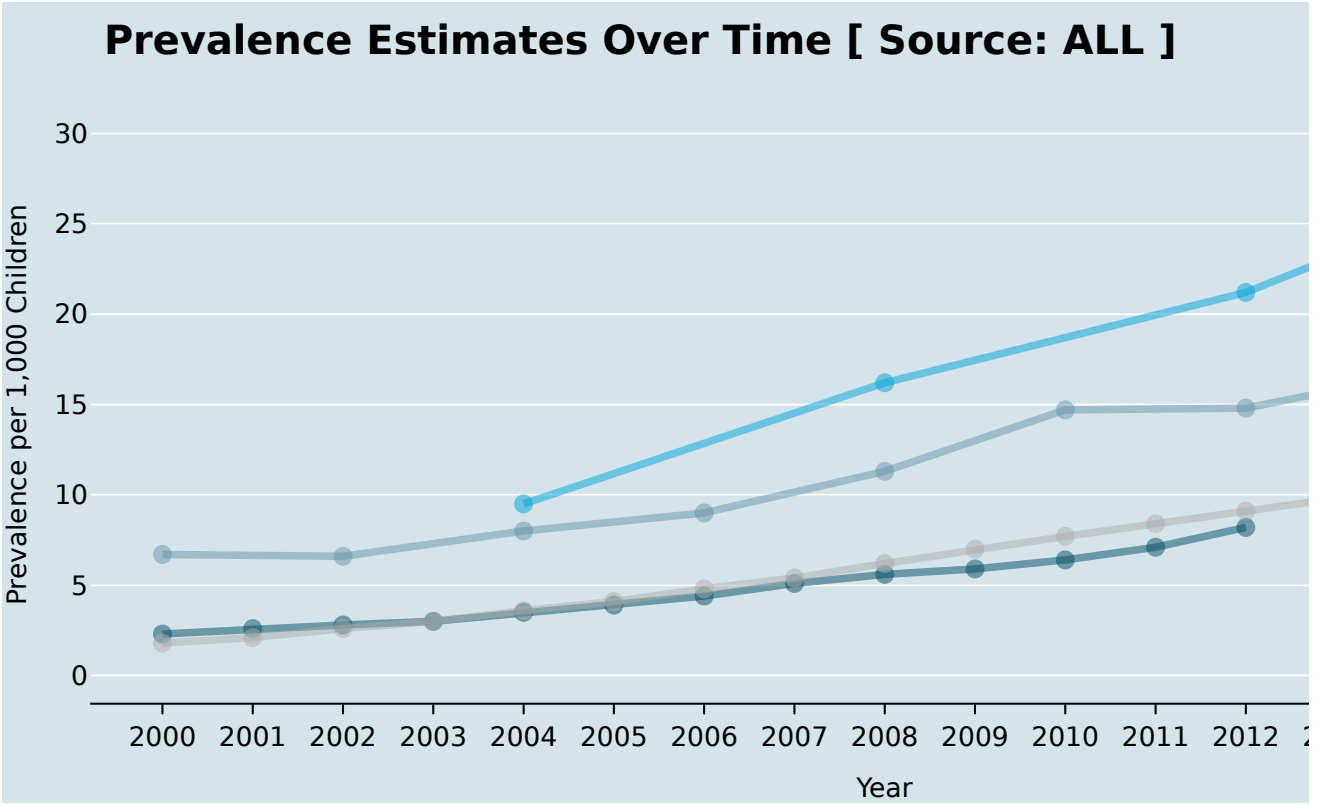
```
Scale for 'colour' is already present. Adding another scale for 'colour',
which will replace the existing scale.
```



Dynamic chart with theme of the economist magazine:

In [195]:
```
# Dynamic chart with theme of the economist magazine:
p_dynamic <- p + theme_economist() + scale_colour_economist()
p_dynamic <- ggplotly(p_dynamic)
p_dynamic
```

```
Scale for 'colour' is already present. Adding another scale for 'colour',
which will replace the existing scale.
```



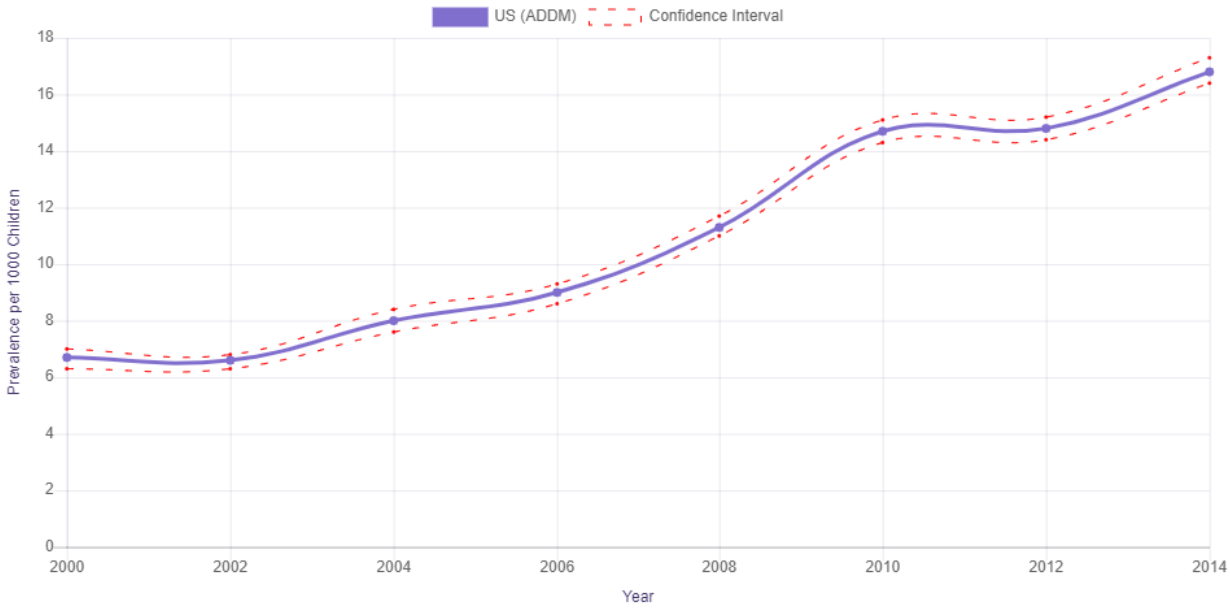
Data Visualisation (Enhanced) - [CDC] ADDM Network estimates for overall ASD prevalence in US over time [Source: ADDM] over [Year]

ADDM NETWORK DATA

In this section, explore the most recent ADDM data, both overall and among certain demographic groups by study area.

ADDM Network estimates for overall ASD prevalence in US over time

with confidence interval



**ADDM data do not represent the entire state, only a selection of sites within the state.*

***ADDM estimate = the total for all sites combined.*

†NSCH data are not comparable over time as data collection methods changed and the data are not provided here. See technical notes for further details.

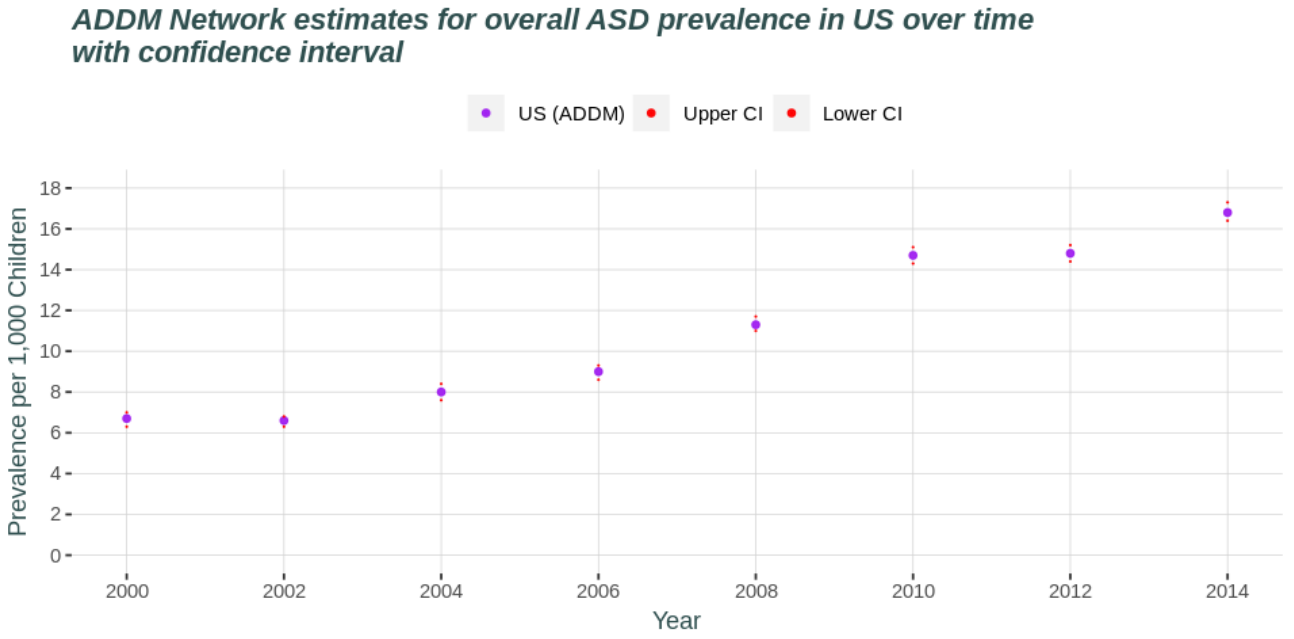
Data Visualisation (Enhanced) - [R] ADDM Network estimates for overall ASD prevalence in US over time [Source: ADDM] over [Year]

In [196]:
```r
# Adjust in-line plot size to M x N
options(repr.plot.width=8, repr.plot.height=4)
```

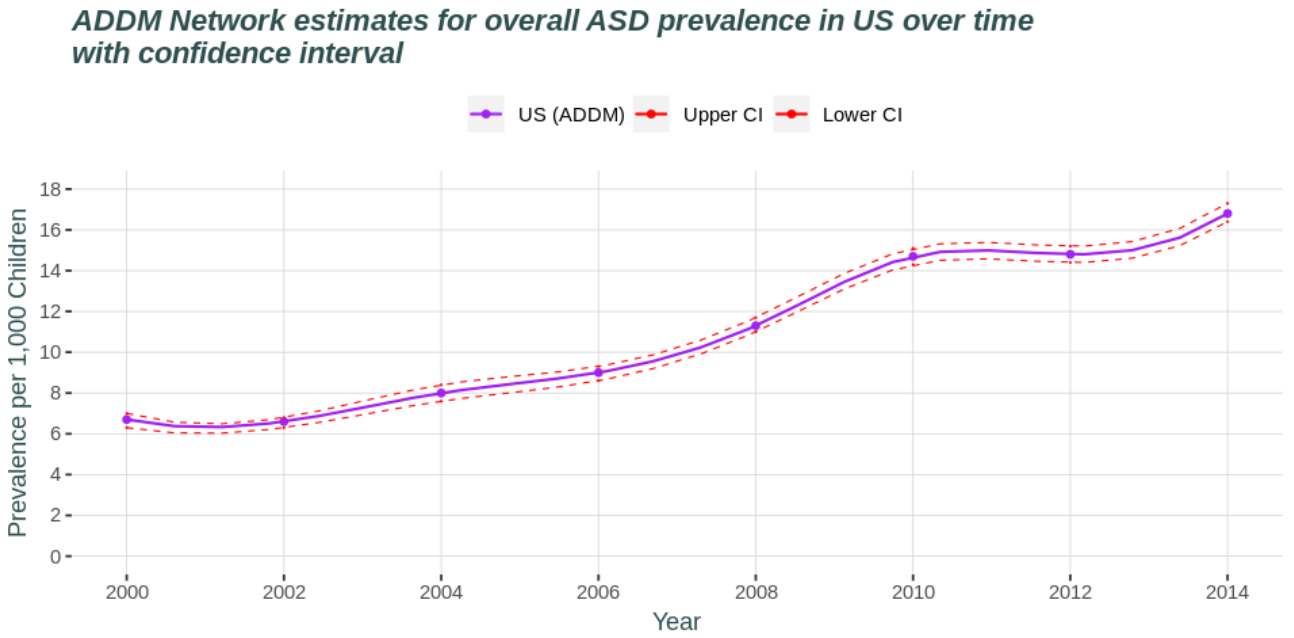
In [197]:
```r
# Filter only data of ADDM
ASD_National_ADDM <- subset(ASD_National, Source == 'addm')
```

In [198]:

```
# ---------------------------------
# [addm] ADDM Network estimates for overall ASD prevalence in US over time
# ---------------------------------

#  Color:
# 'ADDM_Average' "purple"

p <- ggplot(ASD_National_ADDM, aes(x = Year, y = Prevalence)) +
  geom_point(aes(y = Prevalence, color = 'ADDM_Average'), # Name for manual co
             size=2,
             shape=20,
             alpha=0.95) +
  # Add point for Upper.CI
  geom_point(aes(y = Upper.CI, color = 'ADDM_U_CI'), # Name for manual colour
             size=0.1,
             shape=20,
             alpha=0.95) +
  # Add point for Upper.CI
  geom_point(aes(y = Lower.CI, color = 'ADDM_L_CI'), # Name for manual colour
             size=0.1,
             shape=20,
             alpha=0.95) +
  scale_colour_manual(name="",
                      labels = c("US (ADDM)", "Upper CI", "Lower CI"), # Names
                      values = c(ADDM_Average="purple", ADDM_U_CI="red", ADDM_
# Add title, axis label, and axis scale
p <- p + scale_y_continuous(name = "Prevalence per 1,000 Children",
                            breaks = seq(0, 18, 2),
                            limits=c(0, 18)) +
  scale_x_continuous(name = "Year",
                     breaks = seq(2000, 2014, 2),
                     limits = c(2000, 2014)) +
  ggtitle("ADDM Network estimates for overall ASD prevalence in US over time\n
  theme(title = element_text(face = 'bold.italic', color = "darkslategrey"),
        axis.title = element_text(face = 'plain', color = "darkslategrey"),
        panel.background = element_blank(), # Remove chart backgroun colour
        legend.position = 'top',
        panel.grid.major = element_line(size = 0.2, linetype = 'solid', colour
       )
# Show plot
p
```



In [199]:
```
# Add smooth curve to go through date points, using interpolation with splines
# https://stackoverflow.com/questions/35205795/plotting-smooth-line-through-al
spline_ADDM_Prevalence <- as.data.frame(spline(ASD_National_ADDM$Year, ASD_Nat
spline_ADDM_Prevalence_U_CI <- as.data.frame(spline(ASD_National_ADDM$Year, AS
spline_ADDM_Prevalence_L_CI <- as.data.frame(spline(ASD_National_ADDM$Year, AS
# Show plot
p + geom_line(data = spline_ADDM_Prevalence, aes(x = x, y = y, color = 'ADDM_A
  geom_line(data = spline_ADDM_Prevalence_U_CI, aes(x = x, y = y, color = 'ADD
  geom_line(data = spline_ADDM_Prevalence_L_CI, aes(x = x, y = y, color = 'ADD
```



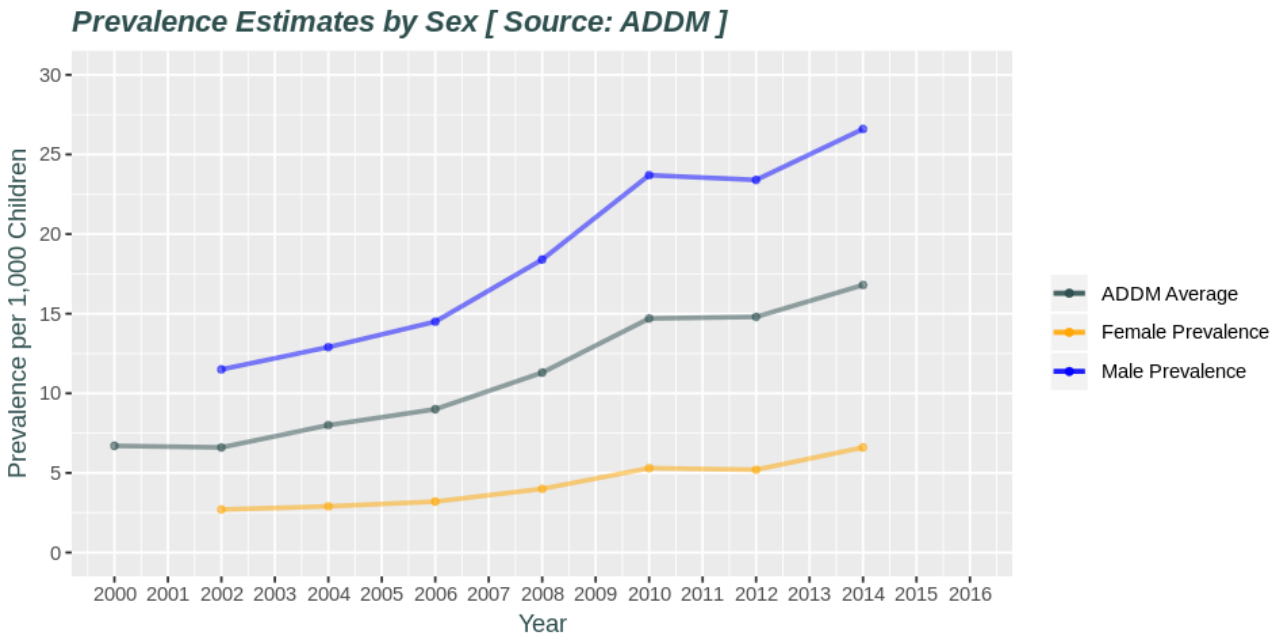
Data Visualisation (Enhanced) - [R] REPORTED PREVALENCE VARIES BY SEX [Source: ADDM] over [Year]

In [200]:
```
# Adjust in-line plot size to M x N
# options(repr.plot.width=8, repr.plot.height=4)
```

In [201]:

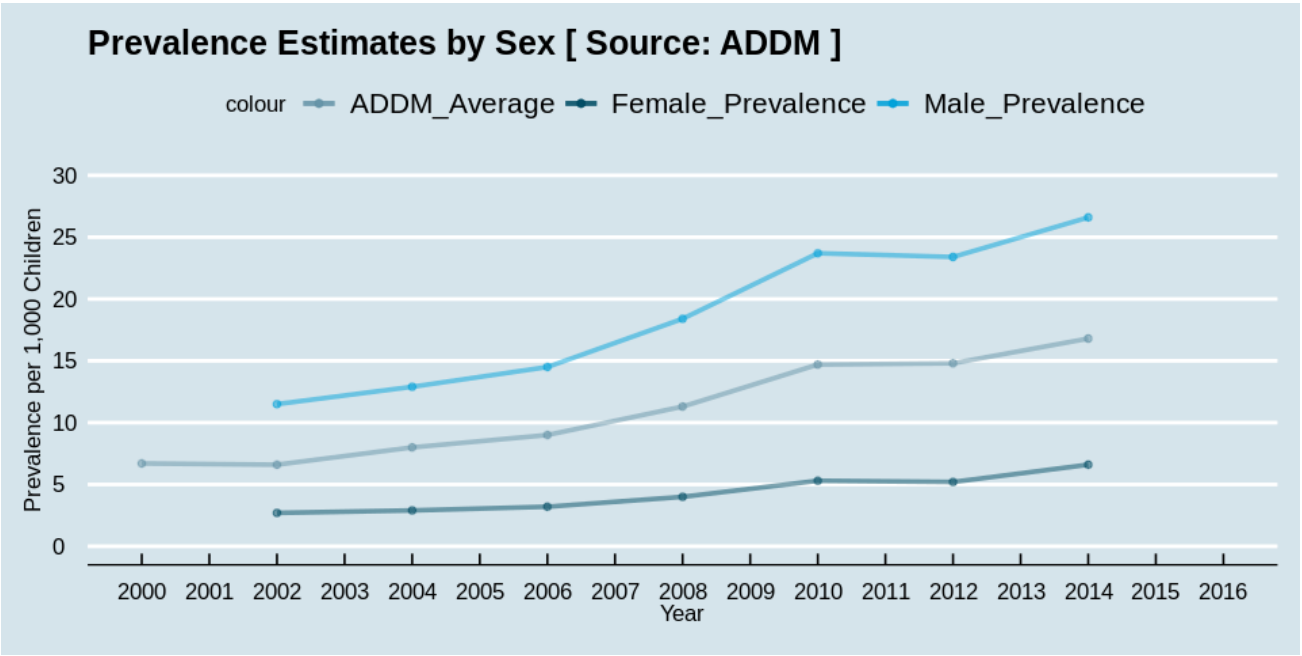
```
# ---------------------------------
# [addm] < Prevalence Varies by Sex >
# ---------------------------------

#  Color:
# 'ADDM_Average' "darkslategrey"
# 'Female_Prevalence' "orange"
# 'Male_Prevalence' "blue"

p <- ggplot(ASD_National_ADDM, aes(x = Year, y = Prevalence)) +
  geom_line(aes(y = Prevalence, colour = 'ADDM_Average'),
            linetype = "solid",  # http://sape.inf.usi.ch/quick-reference/ggpl
            size=1,
            alpha=0.5) +
  geom_point(aes(y = Prevalence, color = 'ADDM_Average'),
             size=2,
             shape=20,
             alpha=0.5) +
  # Add line for Female
  geom_line(aes(y = Female.Prevalence, colour = 'Female_Prevalence'),
            linetype = "solid",  # http://sape.inf.usi.ch/quick-reference/ggpl
            size=1,
            alpha=0.5) +
  geom_point(aes(y = Female.Prevalence, color = 'Female_Prevalence'),
             size=2,
             shape=20,
             alpha=0.5) +
  # Add line for Male
  geom_line(aes(y = Male.Prevalence, colour = 'Male_Prevalence'),
            linetype = "solid",  # http://sape.inf.usi.ch/quick-reference/ggpl
            size=1,
            alpha=0.5) +
  geom_point(aes(y = Male.Prevalence, color = 'Male_Prevalence'),
             size=2,
             shape=20,
             alpha=0.5) +
  scale_colour_manual(name="",
                      labels = c("ADDM Average", "Female Prevalence", "Male Pr
                      values = c(ADDM_Average="darkslategrey", Female_Prevalen
# Add title, axis label, and axis scale
p <- p + scale_y_continuous(name = "Prevalence per 1,000 Children",
                            breaks = seq(0, 30, 5),
                            limits=c(0, 30)) +
  scale_x_continuous(name = "Year",
                     breaks = seq(2000, 2016, 1),
                     limits = c(2000, 2016)) +
  ggtitle("Prevalence Estimates by Sex [ Source: ADDM ]") +
  theme(title = element_text(face = 'bold.italic', color = "darkslategrey"),
        axis.title = element_text(face = 'plain', color = "darkslategrey"))
# Show plot
p
```

```
Warning message:
“Removed 1 rows containing missing values (geom_path).”Warning message:
“Removed 1 rows containing missing values (geom_point).”Warning message:
“Removed 1 rows containing missing values (geom_path).”Warning message:
“Removed 1 rows containing missing values (geom_point).”
```



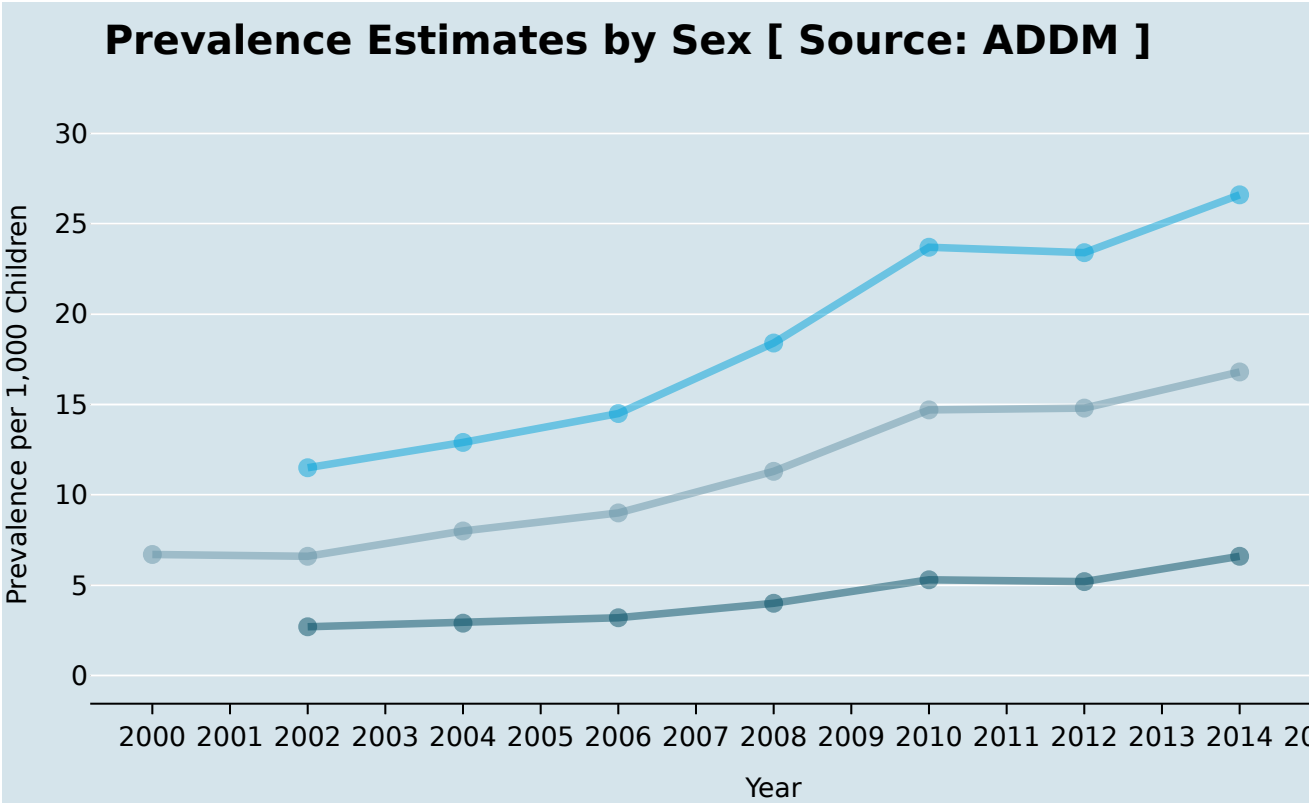
In [202]:
```
# Apply theme
p + theme_economist() + scale_colour_economist() # p + theme_wsj() + scale_col
```

```
Scale for 'colour' is already present. Adding another scale for 'colour',
which will replace the existing scale.
Warning message:
“Removed 1 rows containing missing values (geom_path).”Warning message:
“Removed 1 rows containing missing values (geom_point).”Warning message:
“Removed 1 rows containing missing values (geom_path).”Warning message:
“Removed 1 rows containing missing values (geom_point).”
```



In [203]:
```
# Dynamic chart:
p_dynamic <- p + theme_economist() + scale_colour_economist()
p_dynamic <- ggplotly(p_dynamic)
p_dynamic
```

```
Scale for 'colour' is already present. Adding another scale for 'colour',
which will replace the existing scale.
```



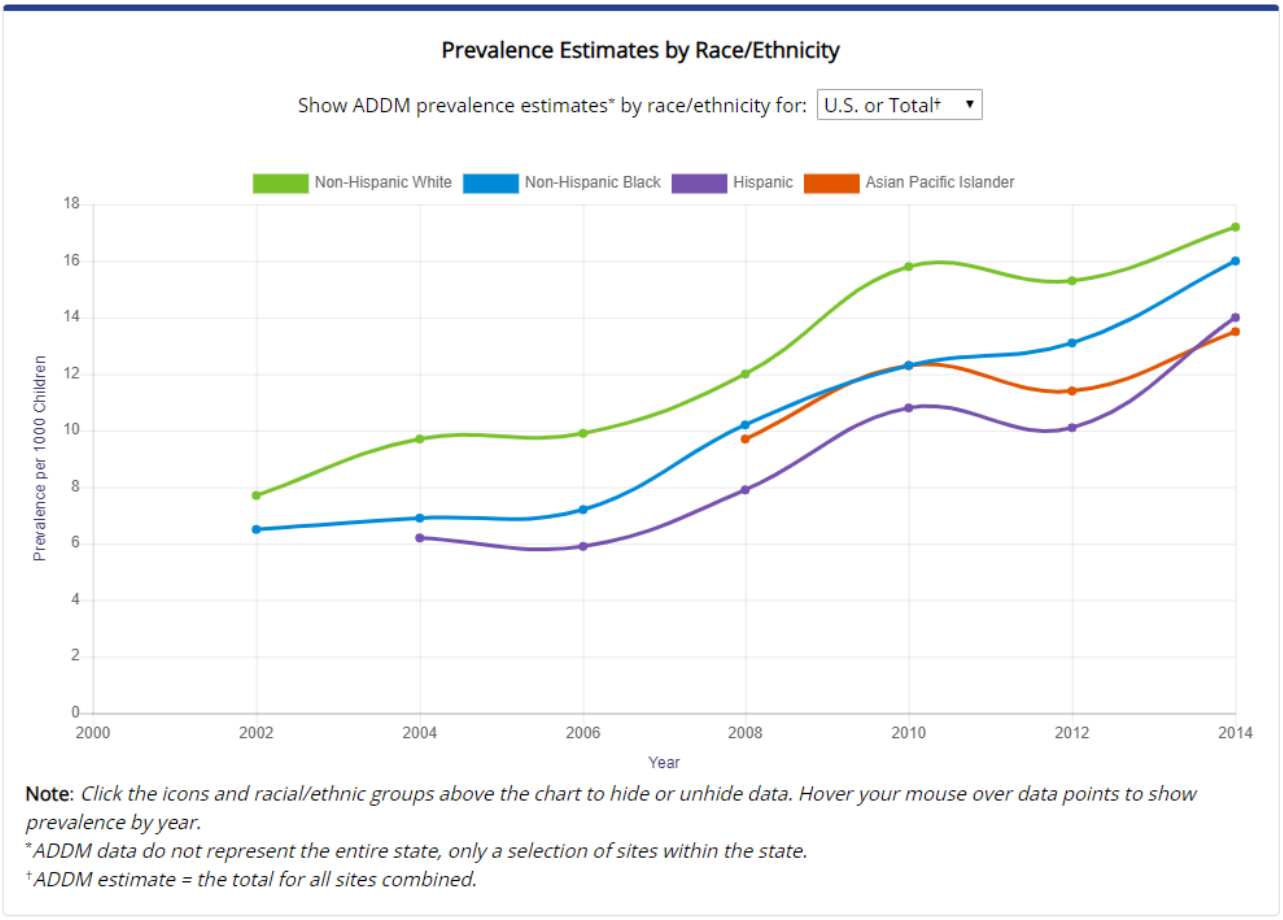
Quiz:

Add 95% Confidence Interval to above plot (Use ggplot)

In [204]:
```
# Write your code below and press Shift+Enter to execute
```

Double-click **here** for the solution.

Data Visualisation (Enhanced) - [CDC] REPORTED PREVALENCE VARIES BY RACE AND ETHNICITY



Note: *Click the icons and racial/ethnic groups above the chart to hide or unhide data. Hover your mouse over data points to show prevalence by year.*

**ADDM data do not represent the entire state, only a selection of sites within the state.*

†ADDM estimate = the total for all sites combined.

Data Visualisation (Enhanced) - [R] REPORTED PREVALENCE VARIES BY RACE AND ETHNICITY [Source: ADDM] With Average

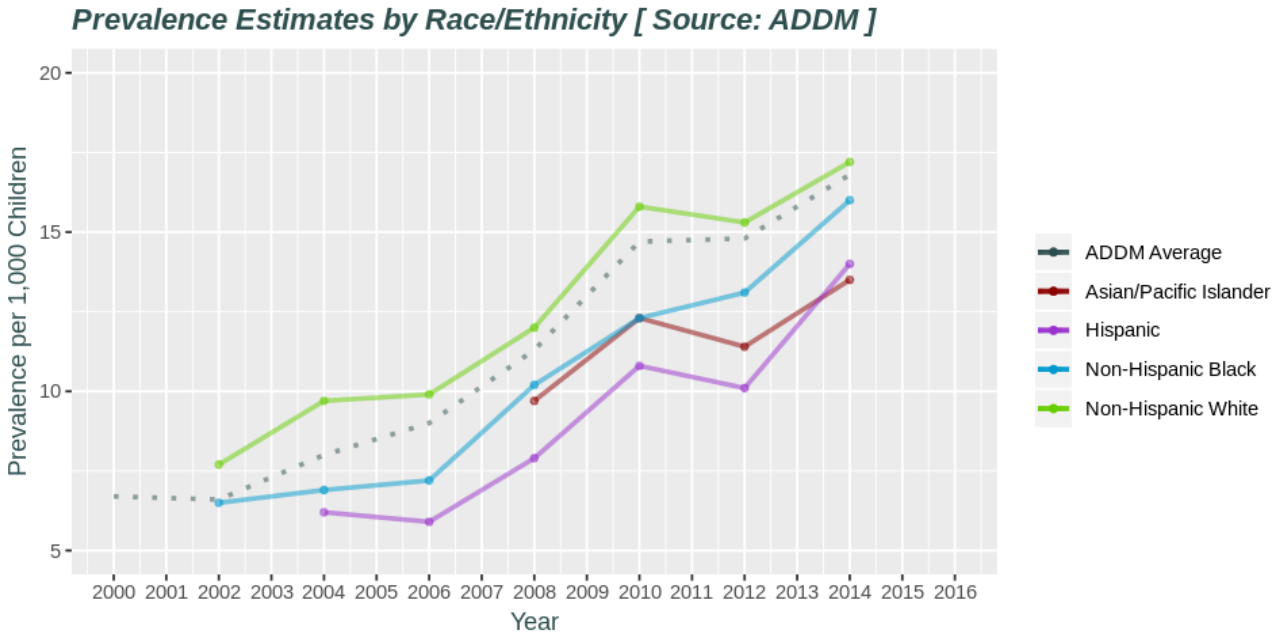
In [205]:
```
# Adjust in-line plot size to M x N
# options(repr.plot.width=8, repr.plot.height=4)
```

In [206]:

```
# ---------------------------------
# [addm] < Prevalence Varies by Race and Ethnicity >
# ---------------------------------

#  Color:
# 'ADDM_Average' "darkslategrey"
# 'Asian_Pacific_Islander' "darkred"
# 'Hispanic' "darkorchid3"
# 'Non_Hispanic_Black' "deepskyblue3"
# 'Non_Hispanic_White' "chartreuse3"

p <- ggplot(ASD_National_ADDM, aes(x = Year, y = Prevalence)) +
  geom_line(aes(y = Prevalence, colour = 'ADDM_Average'),
            linetype = "dotted",  # http://sape.inf.usi.ch/quick-reference/ggp
            size=1,
            alpha=0.5) +
  geom_point(aes(y = Prevalence, color = 'ADDM_Average'),
             size=2,
             shape=20,
             alpha=0) +
  # Add line for Asian.or.Pacific.Islander.Prevalence
  geom_line(aes(y = Asian.or.Pacific.Islander.Prevalence, colour = 'Asian_Paci
            linetype = "solid",  # http://sape.inf.usi.ch/quick-reference/ggpl
            size=1,
            alpha=0.5) +
  geom_point(aes(y = Asian.or.Pacific.Islander.Prevalence, colour = 'Asian_Pac
             size=2,
             shape=20,
             alpha=0.5) +
  # Add line for Hispanic.Prevalence
  geom_line(aes(y = Hispanic.Prevalence, colour = 'Hispanic'),
            linetype = "solid",  # http://sape.inf.usi.ch/quick-reference/ggpl
            size=1,
            alpha=0.5) +
  geom_point(aes(y = Hispanic.Prevalence, colour = 'Hispanic'),
             size=2,
             shape=20,
             alpha=0.5) +
  # Add line for Non.hispanic.black.Prevalence
  geom_line(aes(y = Non.hispanic.black.Prevalence, colour = 'Non_Hispanic_Blac
            linetype = "solid",  # http://sape.inf.usi.ch/quick-reference/ggpl
            size=1,
            alpha=0.5) +
  geom_point(aes(y = Non.hispanic.black.Prevalence, colour = 'Non_Hispanic_Bla
             size=2,
             shape=20,
             alpha=0.5) +
  # Add line for Non.hispanic.white.Prevalence
  geom_line(aes(y = Non.hispanic.white.Prevalence, colour = 'Non_Hispanic_Whit
            linetype = "solid",  # http://sape.inf.usi.ch/quick-reference/ggpl
            size=1,
            alpha=0.5) +
  geom_point(aes(y = Non.hispanic.white.Prevalence, colour = 'Non_Hispanic_Whi
             size=2,
             shape=20,
             alpha=0.5) +
  scale_colour_manual(name="",
                      labels = c("ADDM Average",
                                 "Asian/Pacific Islander",
                                 "Hispanic",
                                 "Non-Hispanic Black",
                                 "Non-Hispanic White"),
                      values = c(ADDM_Average="darkslategrey",
                                 Asian_Pacific_Islander ="darkred",
                                 Hispanic ="darkorchid3",
```

```
                                  Non_Hispanic_Black ="deepskyblue3",
                                  Non_Hispanic_White ="chartreuse3"))
# Add title, axis label, and axis scale
p <- p + scale_y_continuous(name = "Prevalence per 1,000 Children",
                            breaks = seq(5, 20, 5),
                            limits=c(5, 20)) +
  scale_x_continuous(name = "Year",
                     breaks = seq(2000, 2016, 1),
                     limits = c(2000, 2016)) +
  ggtitle("Prevalence Estimates by Race/Ethnicity [ Source: ADDM ]") +
  theme(title = element_text(face = 'bold.italic', color = "darkslategrey"),
        axis.title = element_text(face = 'plain', color = "darkslategrey"))
# Show plot
p
```

```
Warning message:
“Removed 4 rows containing missing values (geom_path).”Warning message:
“Removed 4 rows containing missing values (geom_point).”Warning message:
“Removed 2 rows containing missing values (geom_path).”Warning message:
“Removed 2 rows containing missing values (geom_point).”Warning message:
“Removed 1 rows containing missing values (geom_path).”Warning message:
“Removed 1 rows containing missing values (geom_point).”Warning message:
“Removed 1 rows containing missing values (geom_path).”Warning message:
“Removed 1 rows containing missing values (geom_point).”
```

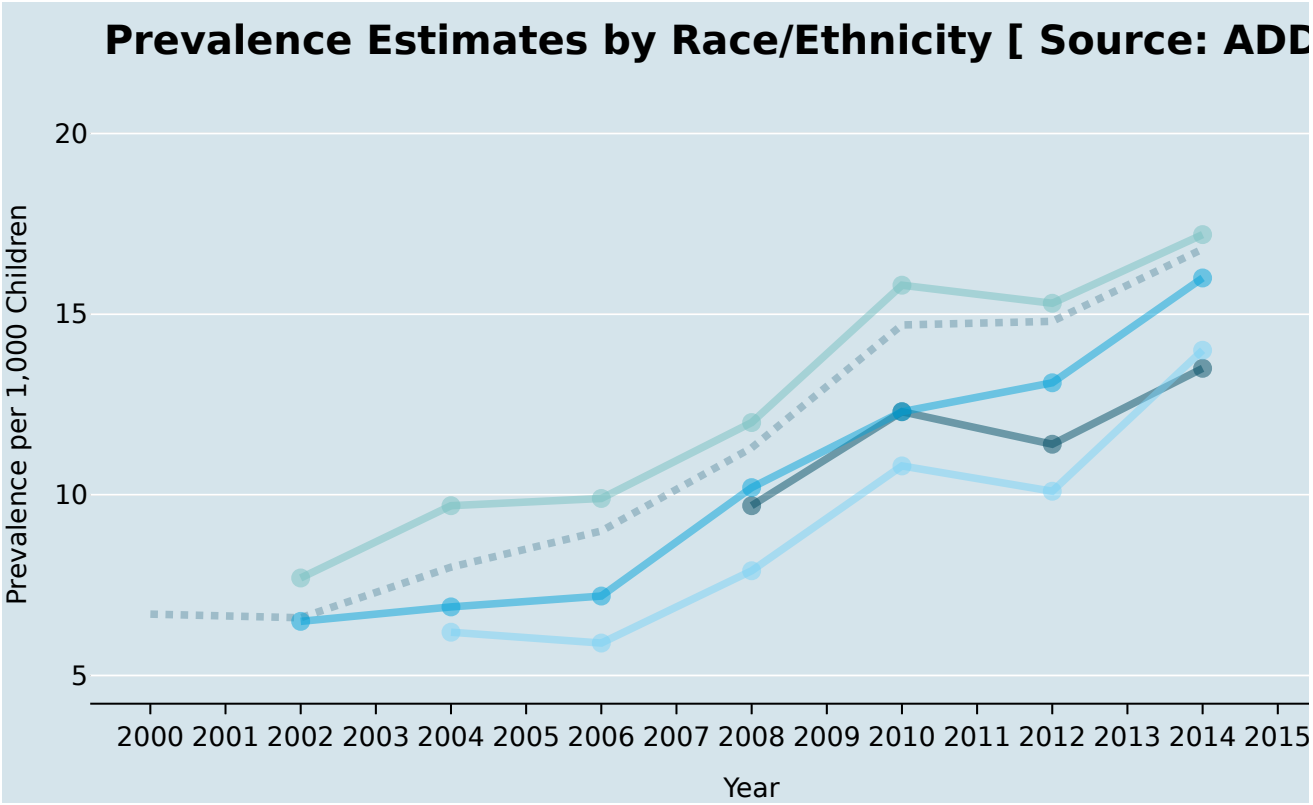


In [207]:

```
# Apply theme
# p + theme_economist() + scale_colour_economist() # p + theme_wsj() + scale_c
```

In [208]:
```
# Dynamic chart:
p_dynamic <- p + theme_economist() + scale_colour_economist()
p_dynamic <- ggplotly(p_dynamic)
p_dynamic
```

```
Scale for 'colour' is already present. Adding another scale for 'colour',
which will replace the existing scale.
```



Quiz:

Change above zig-zag lines to spline/smooth lines.

Hints: Refer to ADDM Network estimates for overall ASD prevalence in US over time.

In [209]:
```
# Write your code below and press Shift+Enter to execute
```

Double-click **here** for the solution.

()

Data Visualisation (Enhanced) - US. State Level Data Processing

In [210]:
```
# ---------------------------------
# Dataset: US. State Level Children ASD Prevalence
# ---------------------------------

ASD_State    <- read.csv("../dataset/ADV_ASD_State.csv", stringsAsFactors = FA

# Obtain number of rows and number of columns/features/variables
dim(ASD_State)
# Obtain overview (data structure/types)
str(ASD_State)
```

```
1692  49

'data.frame':   1692 obs. of  49 variables:
 $ State                                 : chr  "AZ" "GA" "MD" "NJ" ...
 $ Denominator                           : int  45322 43593 21532 29714 24535
23065 35472 45113 36472 11020 ...
 $ Prevalence                            : num  6.5 6.5 5.5 9.9 6.3 4.5 3.3
6.2 6.9 5.9 ...
 $ Lower.CI                              : num  5.8 5.8 4.6 8.9 5.4 3.7 2.7
5.5 6.1 4.6 ...
 $ Upper.CI                              : num  7.3 7.3 6.6 11.1 7.4 5.5 3.9
7 7.8 7.5 ...
 $ Year                                  : int  2000 2000 2000 2000 2000 2000
2002 2002 2002 2002 ...
 $ Source                                : chr  "addm" "addm" "addm" "addm"
...
 $ Source_Full1                          : chr  "Autism & Developmental Disab
ilities Monitoring Network" "Autism & Developmental Disabilities Monitoring N
etwork" "Autism & Developmental Disabilities Monitoring Network" "Autism & De
velopmental Disabilities Monitoring Network" ...
 $ State_Full1                           : chr  "Arizona" "Georgia" "Marylan
d" "New Jersey" ...
 $ State_Full2                           : chr  "AZ-Arizona" "GA-Georgia" "MD
-Maryland" "NJ-New Jersey" ...
 $ Numerator_ASD                         : int  295 283 118 294 155 104 117 2
80 252 65 ...
 $ Numerator_NonASD                      : int  45027 43310 21414 29420 24380
22961 35355 44833 36220 10955 ...
 $ Proportion                            : num  0.00651 0.00649 0.00548 0.009
89 0.00632 ...
 $ X95_Z_CI                              : num  0.00074 0.000754 0.000986 0.0
01125 0.000991 ...
 $ Z_Lower.CI                            : num  5.77 5.74 4.49 8.77 5.33 ...
 $ Z_Upper.CI                            : num  7.25 7.25 6.47 11.02 7.31 ...
 $ Z_Lower.CI_ABSerror                   : num  0.0314 0.062 0.1059 0.1311 0.
0739 ...
 $ Z_Upper.CI_ABSerror                   : num  0.0507 0.0542 0.1337 0.0803
0.0911 ...
 $ Chi_Wilson_P                          : num  0.00655 0.00654 0.00557 0.009
96 0.00639 ...
 $ X95_Chi_Wilson_CI                     : num  0.000741 0.000755 0.00099 0.0
01127 0.000994 ...
 $ Chi_Wilson_Lower.CI                   : num  5.81 5.78 4.58 8.83 5.4 ...
 $ Chi_Wilson_Upper.CI                   : num  7.29 7.29 6.56 11.08 7.39 ...
 $ Chi_Wilson_Lower.CI_ABSerror          : num  0.009314 0.019761 0.021503 0.
069416 0.000453 ...
 $ Chi_Wilson_Upper.CI_ABSerror          : num  0.0077 0.00953 0.04165 0.0152
3 0.01087 ...
 $ Chi_Wilson_Corrected_w_minus.CI       : num  0.0058 0.00577 0.00456 0.0088
1 0.00538 ...
 $ Chi_Wilson_Corrected_w_plus.CI        : num  0.0073 0.0073 0.00658 0.0111
0.00741 ...
 $ Chi_Wilson_Corrected_Lower.CI         : num  5.8 5.77 4.56 8.81 5.38 ...
 $ Chi_Wilson_Corrected_Upper.CI         : num  7.3 7.3 6.58 11.1 7.41 ...
```

```
 $ Chi_Wilson_Corrected_Lower.CI_ABSerror: num  0.00109 0.03057 0.04265 0.085
29 0.01834 ...
 $ Chi_Wilson_Corrected_Upper.CI_ABSerror: num  0.00395 0.0026 0.01636 0.0025
4 0.01108 ...
 $ Male.Prevalence                       : num  9.7 11 8.6 14.8 9.3 6.6 5 10.
1 10.7 9.9 ...
 $ Male.Lower.CI                         : num  8.5 9.7 7.1 13 7.8 5.2 4.1 8.
8 9.3 7.6 ...
 $ Male.Upper.CI                         : num  11.1 12.4 10.6 16.8 11.2 8.2
6.2 11.4 12.3 12.9 ...
 $ Female.Prevalence                     : num  3.2 2 2.2 4.3 3.3 2.4 1.4 2.2
2.9 1.7 ...
 $ Female.Lower.CI                       : num  2.5 1.5 1.5 3.3 2.4 1.6 0.9
1.7 2.2 0.9 ...
 $ Female.Upper.CI                       : num  4 2.7 2.7 5.5 4.5 3.5 2.1 2.9
3.8 3.2 ...
 $ Non.hispanic.white.Prevalence         : num  8.6 7.9 4.9 11.3 6.5 4.5 3.3
7.7 7.4 6.4 ...
 $ Non.hispanic.white.Lower.CI           : num  7.5 6.7 3.8 9.5 5.2 3.7 2.6
6.7 6.5 4.8 ...
 $ Non.hispanic.white.Upper.CI           : num  9.8 9.3 6.4 13.3 8.2 5.5 4.1
8.9 8.6 8.5 ...
 $ Non.hispanic.black.Prevalence         : chr  "7.3" "5.3" "6.1" "10.6" ...
 $ Non.hispanic.black.Lower.CI           : chr  "4.4" "4.4" "4.7" "8.5" ...
 $ Non.hispanic.black.Upper.CI           : chr  "12.2" "6.4" "8" "13.1" ...
 $ Hispanic.Prevalence                   : chr  "No data" "No data" "No data"
"No data" ...
 $ Hispanic.Lower.CI                     : chr  "No data" "No data" "No data"
"No data" ...
 $ Hispanic.Upper.CI                     : chr  "No data" "No data" "No data"
"No data" ...
 $ Asian.or.Pacific.Islander.Prevalence  : chr  "No data" "No data" "No data"
"No data" ...
 $ Asian.or.Pacific.Islander.Lower.CI    : chr  "No data" "No data" "No data"
"No data" ...
 $ Asian.or.Pacific.Islander.Upper.CI    : chr  "No data" "No data" "No data"
"No data" ...
 $ State_Region                          : chr  "D8 Mountain" "D5 South Atlan
tic" "D5 South Atlantic" "D2 Middle Atlantic" ...
```

Data Visualisation (Enhanced) - US. State Level Data Pre-Process data

Pre-Process data: Missing data

In [211]:
```
# Count missing values in dataframe:
sum(is.na(ASD_State)) # No missing data recognised by R (NA)
# Define several offending strings
na_strings <- c("", "No data", "NA", "N A", "N / A", "N/A", "N/ A", "Not Avail
# Replace these defined missing values to R's internal NA
ASD_State = replace_with_na_all(ASD_State, condition = ~.x %in% na_strings)
# Count missing values in dataframe:
sum(is.na(ASD_State))
```

14454

28992

Remove invalid unicode char/string: \x92

In [212]:
```
# Remove invalid unicode char/string: \x92
ASD_State$Source_Full1[ASD_State$Source_Full1 == "National Survey of Children\
```

Delete/Drop variable by index: column from 14 to 26, 29, and 30

In [213]:
```
cbind(names(ASD_State), c(1:length(names(ASD_State))))
```

State	1
Denominator	2
Prevalence	3
Lower.CI	4
Upper.CI	5
Year	6
Source	7
Source_Full1	8
State_Full1	9
State_Full2	10
Numerator_ASD	11
Numerator_NonASD	12
Proportion	13
X95_Z_CI	14
Z_Lower.CI	15
Z_Upper.CI	16
Z_Lower.CI_ABSerror	17
Z_Upper.CI_ABSerror	18
Chi_Wilson_P	19
X95_Chi_Wilson_CI	20
Chi_Wilson_Lower.CI	21
Chi_Wilson_Upper.CI	22
Chi_Wilson_Lower.CI_ABSerror	23
Chi_Wilson_Upper.CI_ABSerror	24
Chi_Wilson_Corrected_w_minus.CI	25
Chi_Wilson_Corrected_w_plus.CI	26
Chi_Wilson_Corrected_Lower.CI	27
Chi_Wilson_Corrected_Upper.CI	28
Chi_Wilson_Corrected_Lower.CI_ABSerror	29
Chi_Wilson_Corrected_Upper.CI_ABSerror	30
Male.Prevalence	31
Male.Lower.CI	32
Male.Upper.CI	33
Female.Prevalence	34
Female.Lower.CI	35
Female.Upper.CI	36
Non.hispanic.white.Prevalence	37
Non.hispanic.white.Lower.CI	38
Non.hispanic.white.Upper.CI	39
Non.hispanic.black.Prevalence	40
Non.hispanic.black.Lower.CI	41

Non.hispanic.black.Upper.CI	42
Hispanic.Prevalence	43
Hispanic.Lower.CI	44
Hispanic.Upper.CI	45
Asian.or.Pacific.Islander.Prevalence	46
Asian.or.Pacific.Islander.Lower.CI	47
Asian.or.Pacific.Islander.Upper.CI	48
State_Region	49

In [214]:
```
# Delete/Drop variable by index: column from 14 to 26, 29, and 30
# names(ASD_State)
ASD_State <- ASD_State[ -c(14:26, 29, 30) ]
```

Create new variables

In [215]:
```
# Create one new variable: Source_UC as uppercase of Source
ASD_State$Source_UC <- paste(toupper(ASD_State$Source))
# Create one new variable: Source_Full3 by combining Source_UC and Source_Full
ASD_State$Source_Full3 <- paste(ASD_State$Source_UC, ASD_State$Source_Full1)
```

Create one new ordinal categorical variable: Prevalence_Rank2 ("Low", "High") by binning Prevalence

In [216]:
```
# Recode Risk into category from Prevalence

# Low [0, 5)
# High [5, +oo)

ASD_State$Prevalence_Risk2[ASD_State$Prevalence < 5] = "Low"
ASD_State$Prevalence_Risk2[ASD_State$Prevalence >= 5 ] = "High"
#
# head(ASD_State)
```

```
Warning message:
“Unknown or uninitialised column: 'Prevalence_Risk2'.”
```

Create one new ordinal categorical variable: Prevalence_Rank4 ("Low", "Medium", "High", "Very High") by binning Prevalence

In [217]:
```
# Recode Risk into category from Prevalence

# Low [0, 5)
# Medium [5, 10)
# High [10, 20)
# Very High [20, +oo)

ASD_State$Prevalence_Risk4 = "Very High"
ASD_State$Prevalence_Risk4[ASD_State$Prevalence < 20 ] = "High"
ASD_State$Prevalence_Risk4[ASD_State$Prevalence < 10 ] = "Medium"
ASD_State$Prevalence_Risk4[ASD_State$Prevalence < 5] = "Low"
#
# head(ASD_State)
```

Convert to correct data types

In [218]:
```
str(ASD_State)
```

```
Classes ‘tbl_df’, ‘tbl’ and 'data.frame':       1692 obs. of  38 variables:
 $ State                                : chr  "AZ" "GA" "MD" "NJ" ...
 $ Denominator                          : int  45322 43593 21532 29714 24535 2
3065 35472 45113 36472 11020 ...
 $ Prevalence                           : num  6.5 6.5 5.5 9.9 6.3 4.5 3.3 6.2
6.9 5.9 ...
 $ Lower.CI                             : num  5.8 5.8 4.6 8.9 5.4 3.7 2.7 5.5
6.1 4.6 ...
 $ Upper.CI                             : num  7.3 7.3 6.6 11.1 7.4 5.5 3.9 7
7.8 7.5 ...
 $ Year                                 : int  2000 2000 2000 2000 2000 2000 2
002 2002 2002 2002 ...
 $ Source                               : chr  "addm" "addm" "addm" "addm" ...
 $ Source_Full1                         : chr  "Autism & Developmental Disabil
ities Monitoring Network" "Autism & Developmental Disabilities Monitoring Net
work" "Autism & Developmental Disabilities Monitoring Network" "Autism & Deve
lopmental Disabilities Monitoring Network" ...
 $ State_Full1                          : chr  "Arizona" "Georgia" "Maryland"
"New Jersey" ...
 $ State_Full2                          : chr  "AZ-Arizona" "GA-Georgia" "MD-M
aryland" "NJ-New Jersey" ...
 $ Numerator_ASD                        : int  295 283 118 294 155 104 117 280
252 65 ...
 $ Numerator_NonASD                     : int  45027 43310 21414 29420 24380 2
2961 35355 44833 36220 10955 ...
 $ Proportion                           : num  0.00651 0.00649 0.00548 0.00989
0.00632 ...
 $ Chi_Wilson_Corrected_Lower.CI        : num  5.8 5.77 4.56 8.81 5.38 ...
 $ Chi_Wilson_Corrected_Upper.CI        : num  7.3 7.3 6.58 11.1 7.41 ...
 $ Male.Prevalence                      : num  9.7 11 8.6 14.8 9.3 6.6 5 10.1
10.7 9.9 ...
 $ Male.Lower.CI                        : num  8.5 9.7 7.1 13 7.8 5.2 4.1 8.8
9.3 7.6 ...
 $ Male.Upper.CI                        : num  11.1 12.4 10.6 16.8 11.2 8.2 6.
2 11.4 12.3 12.9 ...
 $ Female.Prevalence                    : num  3.2 2 2.2 4.3 3.3 2.4 1.4 2.2
2.9 1.7 ...
 $ Female.Lower.CI                      : num  2.5 1.5 1.5 3.3 2.4 1.6 0.9 1.7
2.2 0.9 ...
 $ Female.Upper.CI                      : num  4 2.7 2.7 5.5 4.5 3.5 2.1 2.9
3.8 3.2 ...
 $ Non.hispanic.white.Prevalence        : num  8.6 7.9 4.9 11.3 6.5 4.5 3.3 7.
7 7.4 6.4 ...
 $ Non.hispanic.white.Lower.CI          : num  7.5 6.7 3.8 9.5 5.2 3.7 2.6 6.7
6.5 4.8 ...
 $ Non.hispanic.white.Upper.CI          : num  9.8 9.3 6.4 13.3 8.2 5.5 4.1 8.
9 8.6 8.5 ...
 $ Non.hispanic.black.Prevalence        : chr  "7.3" "5.3" "6.1" "10.6" ...
 $ Non.hispanic.black.Lower.CI          : chr  "4.4" "4.4" "4.7" "8.5" ...
 $ Non.hispanic.black.Upper.CI          : chr  "12.2" "6.4" "8" "13.1" ...
 $ Hispanic.Prevalence                  : chr  NA NA NA NA ...
 $ Hispanic.Lower.CI                    : chr  NA NA NA NA ...
 $ Hispanic.Upper.CI                    : chr  NA NA NA NA ...
 $ Asian.or.Pacific.Islander.Prevalence: chr  NA NA NA NA ...
 $ Asian.or.Pacific.Islander.Lower.CI   : chr  NA NA NA NA ...
 $ Asian.or.Pacific.Islander.Upper.CI   : chr  NA NA NA NA ...
 $ State_Region                         : chr  "D8 Mountain" "D5 South Atlanti
c" "D5 South Atlantic" "D2 Middle Atlantic" ...
 $ Source_UC                            : chr  "ADDM" "ADDM" "ADDM" "ADDM" ...
 $ Source_Full3                         : chr  "ADDM Autism & Developmental Di
sabilities Monitoring Network" "ADDM Autism & Developmental Disabilities Moni
toring Network" "ADDM Autism & Developmental Disabilities Monitoring Network"
"ADDM Autism & Developmental Disabilities Monitoring Network" ...
```

```
 $ Prevalence_Risk2                    : chr  "High" "High" "High" "High" ...
 $ Prevalence_Risk4                    : chr  "Medium" "Medium" "Medium" "Med
ium" ...
```

In [219]:
```
# cbind(names(ASD_State), c(1:length(names(ASD_State))))
```

Convert variables to numeric

In [220]:
```
# Convert Prevalence and CIs from categorical/chr to numeric
ix <- 13:33 # define an index
ASD_State[ix] <- lapply(ASD_State[ix], as.numeric)
```

Convert variables to categorical/factor

In [221]:
```
# Convert Source from categorical/chr to categorical/factor
ix <- c(1, 7, 8, 9, 10, 34, 35, 36) # define an index
ASD_State[ix] <- lapply(ASD_State[ix], as.factor)

# Create new ordered factor Year_Factor from Year
ASD_State$Year_Factor <- factor(ASD_State$Year, ordered = TRUE)
```

Convert Prevalence_Rank2 & Prevalence_Rank4 to ordered factor

In [222]:
```
# Convert to factor
ASD_State$Prevalence_Risk2 = factor(ASD_State$Prevalence_Risk2, ordered=TRUE,
                                    levels=c("Low", "High"))
# Convert to factor
ASD_State$Prevalence_Risk4 = factor(ASD_State$Prevalence_Risk4, ordered=TRUE,
                                    levels=c("Low", "Medium", "High", "
```

In [223]:
```
# Display unique values (levels) of a factor categrotical
lapply(select_if(ASD_State, is.factor), levels)
```

$State
'AK' 'AL' 'AR' 'AZ' 'CA' 'CO' 'CT' 'DC' 'DE' 'FL' 'GA' 'HI' 'IA' 'ID' 'IL' 'IN' 'KS' 'KY' 'LA' 'MA' 'MD' 'ME' 'MI' 'MN' 'MO' 'MS' 'MT' 'NC' 'ND' 'NE' 'NH' 'NJ' 'NM' 'NV' 'NY' 'OH' 'OK' 'OR' 'PA' 'RI' 'SC' 'SD' 'TN' 'TX' 'UT' 'VA' 'VT' 'WA' 'WI' 'WV' 'WY'

$Source
'addm' 'medi' 'nsch' 'sped'

$Source_Full1
'Autism & Developmental Disabilities Monitoring Network' 'Medicaid'
'National Survey of Children\'s Health' 'Special Education Child Count'

$State_Full1
'Alabama' 'Alaska' 'Arizona' 'Arkansas' 'California' 'Colorado' 'Connecticut' 'Delaware' 'District of Columbia' 'Florida' 'Georgia' 'Hawaii' 'Idaho' 'Illinois' 'Indiana' 'Iowa' 'Kansas' 'Kentucky' 'Louisiana' 'Maine' 'Maryland' 'Massachusetts' 'Michigan' 'Minnesota' 'Mississippi' 'Missouri' 'Montana' 'Nebraska' 'Nevada' 'New Hampshire' 'New Jersey' 'New Mexico' 'New York' 'North Carolina' 'North Dakota' 'Ohio' 'Oklahoma' 'Oregon' 'Pennsylvania' 'Rhode Island' 'South Carolina' 'South Dakota' 'Tennessee' 'Texas' 'Utah' 'Vermont' 'Virginia' 'Washington' 'West Virginia' 'Wisconsin' 'Wyoming'

$State_Full2
'AK-Alaska' 'AL-Alabama' 'AR-Arkansas' 'AZ-Arizona' 'CA-California' 'CO-Colorado' 'CT-Connecticut' 'DC-District of Columbia' 'DE-Delaware' 'FL-Florida' 'GA-Georgia' 'HI-Hawaii' 'IA-Iowa' 'ID-Idaho' 'IL-Illinois' 'IN-Indiana' 'KS-Kansas' 'KY-Kentucky' 'LA-Louisiana' 'MA-Massachusetts' 'MD-Maryland' 'ME-Maine' 'MI-Michigan' 'MN-Minnesota' 'MO-Missouri' 'MS-Mississippi' 'MT-Montana' 'NC-North Carolina' 'ND-North Dakota' 'NE-Nebraska' 'NH-New Hampshire' 'NJ-New Jersey' 'NM-New Mexico' 'NV-Nevada' 'NY-New York' 'OH-Ohio' 'OK-Oklahoma' 'OR-Oregon' 'PA-Pennsylvania' 'RI-Rhode Island' 'SC-South Carolina' 'SD-South Dakota' 'TN-Tennessee' 'TX-Texas' 'UT-Utah' 'VA-Virginia' 'VT-Vermont' 'WA-Washington' 'WI-Wisconsin' 'WV-West Virginia' 'WY-Wyoming'

$State_Region
'D1 New England' 'D2 Middle Atlantic' 'D3 East North Central' 'D4 West North Central' 'D5 South Atlantic' 'D6 East South Central' 'D7 West South Central' 'D8 Mountain' 'D9 Pacific'

$Source_UC
'ADDM' 'MEDI' 'NSCH' 'SPED'

$Source_Full3
'ADDM Autism & Developmental Disabilities Monitoring Network' 'MEDI Medicaid'
'NSCH National Survey of Children\'s Health' 'SPED Special Education Child Count'

$Prevalence_Risk2
'Low' 'High'

$Prevalence_Risk4
'Low' 'Medium' 'High' 'Very High'

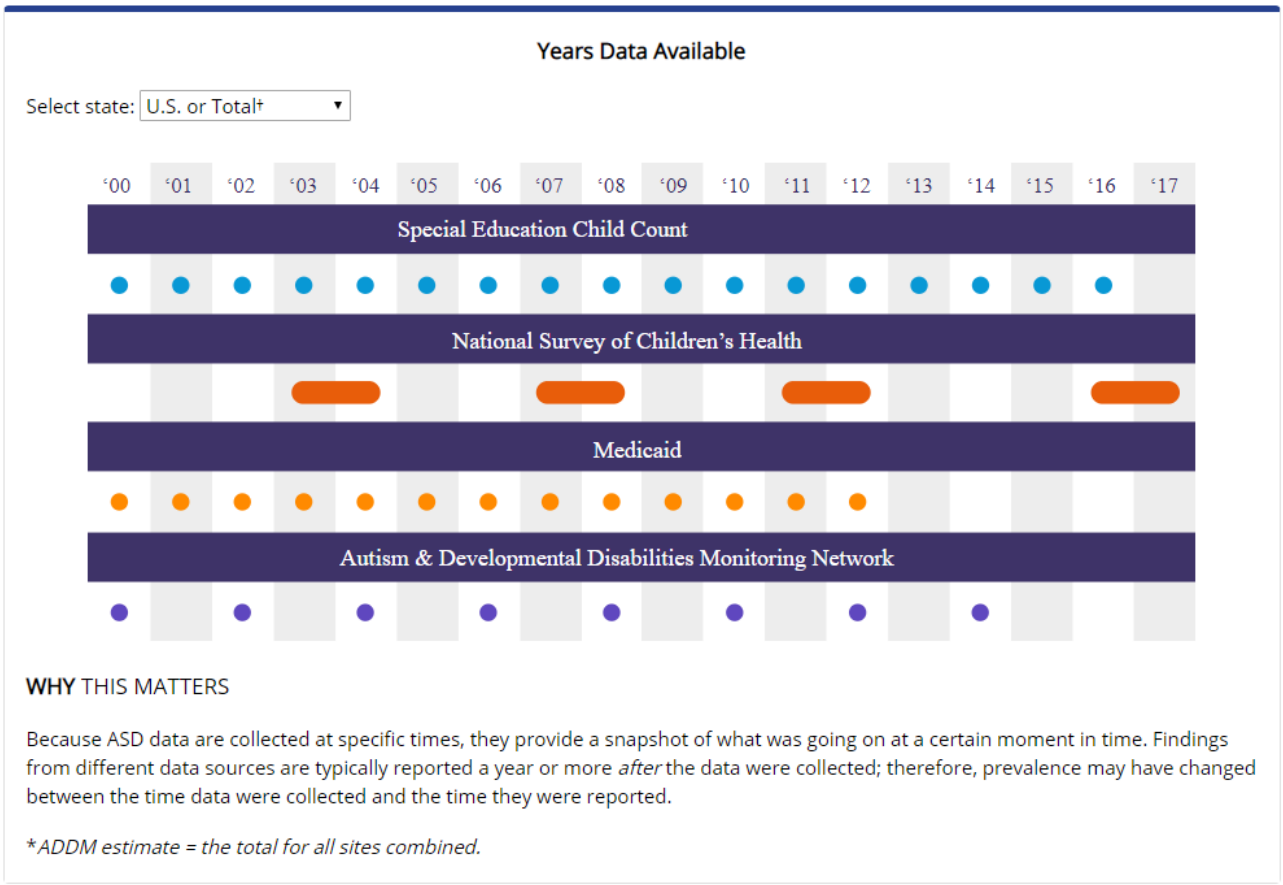
$Year_Factor
'2000' '2001' '2002' '2003' '2004' '2005' '2006' '2007' '2008' '2009' '2010' '2011' '2012' '2013' '2014' '2015' '2016'

Optionally, export the processed dataframe data to CSV file.

In [224]:
```
write.csv(ASD_State, file = "../dataset/ADV_ASD_State_R.csv", row.names = FALS
```

In [225]:
```
# Read back in above saved file:
# ASD_State <- read.csv("../dataset/ADV_ASD_State_R.csv")
# ASD_State$Year_Factor <- factor(ASD_State$Year_Factor, ordered = TRUE) # Con
# ASD_State$Prevalence_Risk2 = factor(ASD_State$Prevalence_Risk2, ordered=TRUE
# ASD_State$Prevalence_Risk4 = factor(ASD_State$Prevalence_Risk4, ordered=TRUE
```

Data Visualisation (Enhanced) - US. State Level Data Visualisation



Above chat shows at data source level, we'd also like to know State level data availbility. How?

Data Visualisation (Enhanced) - [R] Explore the Data [Years Data Available by State]

In [226]:
```
# Adjust in-line plot size to M x N
options(repr.plot.width=8, repr.plot.height=12)
```

In [227]:

```
# ---------------------------------
# [State] < Years Data Available by State >
# ---------------------------------
p <- ggplot(ASD_State, aes(x = Source, fill = Source)) +
  geom_bar() + theme(axis.text.x=element_blank(),  # Hide axis
                     axis.ticks.x=element_blank(), # Hide axis
                     axis.text.y=element_blank(),  # Hide axis
                     axis.ticks.y=element_blank(), # Hide axis
                     panel.background = element_blank(), # Remove panel backgr
                     legend.position="top",
                     strip.text.y = element_text(angle=0) # Rotate text to hor
  ) +
  scale_fill_manual("Data Source:", values = c("addm" = "darkblue",
                                               "medi" = "orange",
                                               "nsch" = "darkred",
                                               "sped" = "skyblue")) +
  facet_grid(facets = State_Full2 ~ Year) +
  labs(x="", y="", title="Years Data Available by State") # layers of graphics
```

In [228]:
```
# Below plot may run for a while
# Show plot
p
```

Filter and create dataframe of different data sources, for easy data access

In [229]:
```
# Filter and create dataframe of different data sources, for easy data access
ASD_State_ADDM <- subset(ASD_State, Source == 'addm')
ASD_State_MEDI <- subset(ASD_State, Source == 'medi')
ASD_State_NSCH <- subset(ASD_State, Source == 'nsch')
ASD_State_SPED <- subset(ASD_State, Source == 'sped')
```

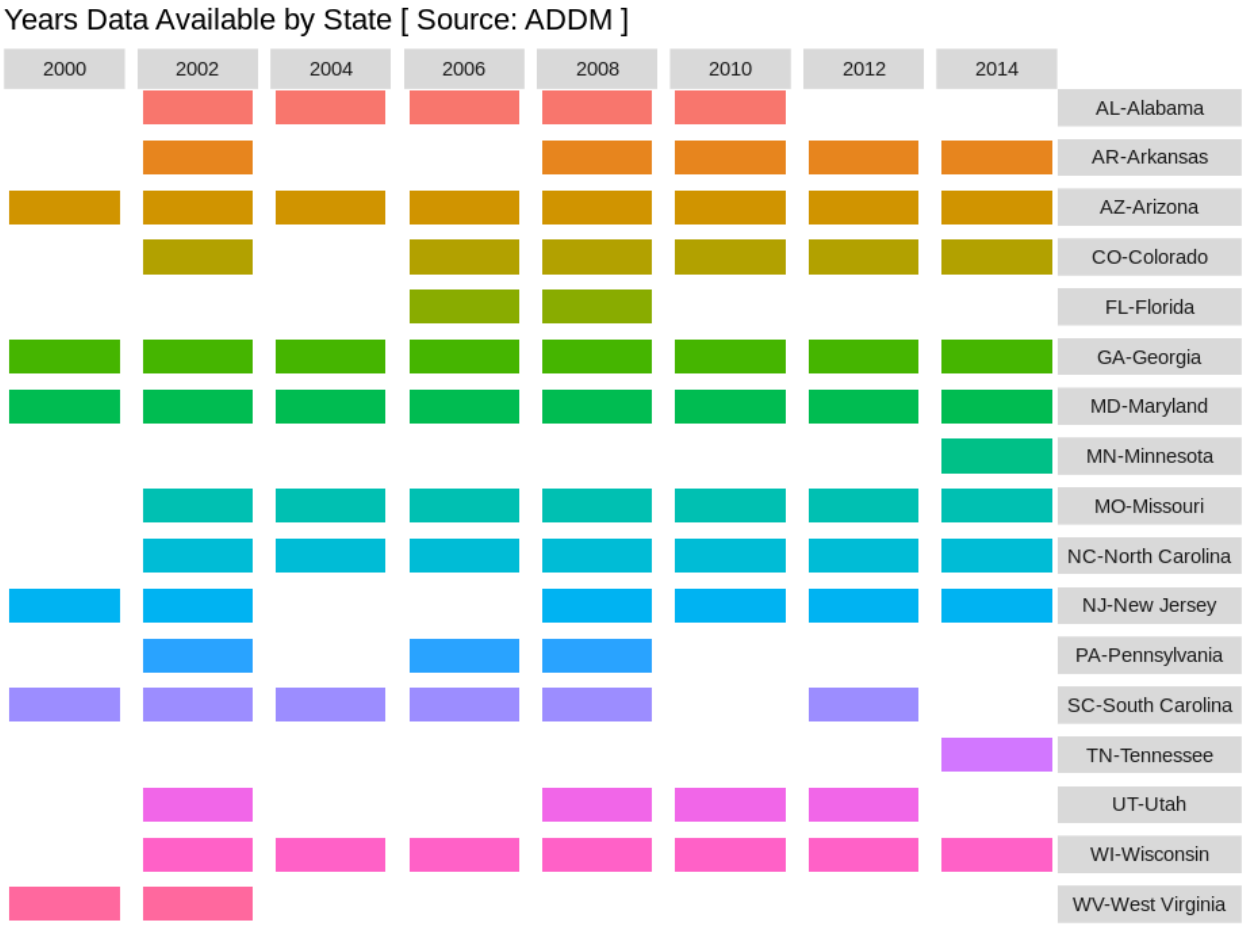
Data Visualisation (Enhanced) - [R] Explore the Data Years Data Available by State [Source: ADDM]

In [230]:
```
# Adjust in-line plot size to M x N
options(repr.plot.width=8, repr.plot.height=6)
```

Years Data Available by State [Source: ADDM]

In [231]:
```
# Years Data Available by State [ Source: ADDM ]
p <- ggplot(ASD_State_ADDM, aes(x = 1, fill = State_Full2)) +
  geom_bar() + theme(axis.text.x=element_blank(),  # Hide axis
                     axis.ticks.x=element_blank(), # Hide axis
                     axis.text.y=element_blank(),  # Hide axis
                     axis.ticks.y=element_blank(), # Hide axis
                     panel.background = element_blank(), # Remove panel backgr
                     legend.position="none",
                     strip.text.y = element_text(angle=0) # Rotate text to hor
  ) +
  facet_grid(facets = State_Full2 ~ Year_Factor) +
  labs(x="", y="", title="Years Data Available by State [ Source: ADDM ]") # l
```

In [232]:
```
# Show plot
p
```



Quiz:

Create Years Data Available by State [Source: XXXX] for other three data sources:

In [233]:
```
# Write your code below and press Shift+Enter to execute
```

Double-click **here** for the solution.

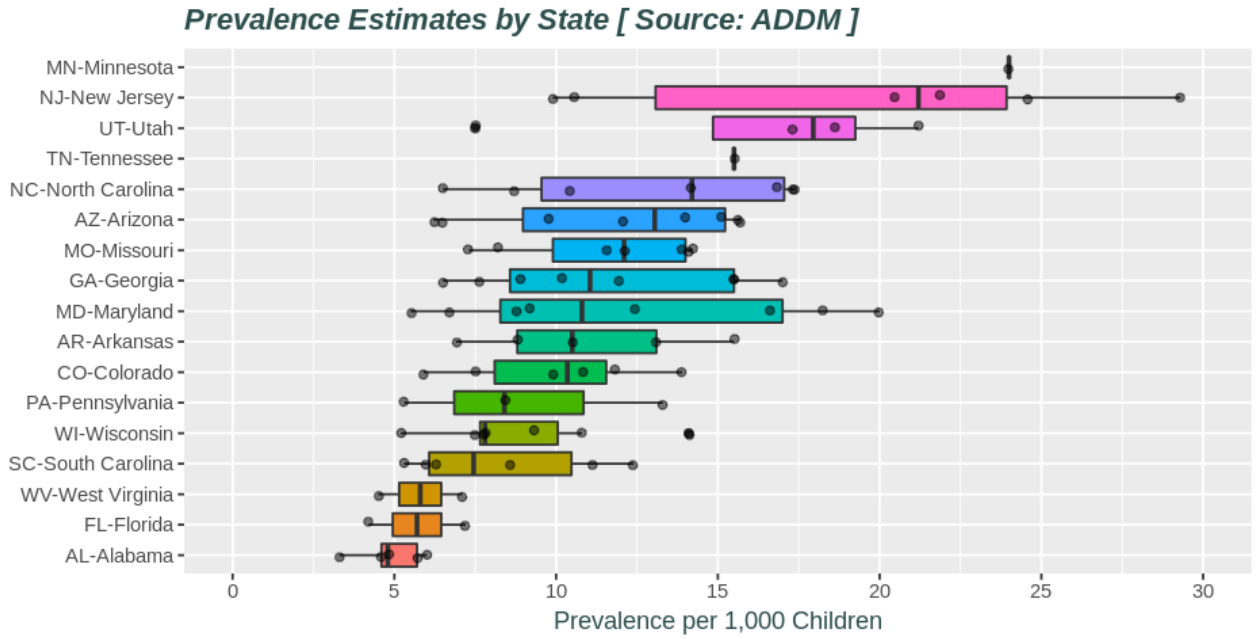
Data Visualisation (Enhanced) - [R] REPORTED PREVALENCE VARIES BY GEOGRAPHIC LOCATION (States) Prevalence Estimates by State [Source: ADDM]

In [234]:
```
# Adjust in-line plot size to M x N
options(repr.plot.width=8, repr.plot.height=4)
```

Visualise: **Prevalence Estimates by State [Source: ADDM]**

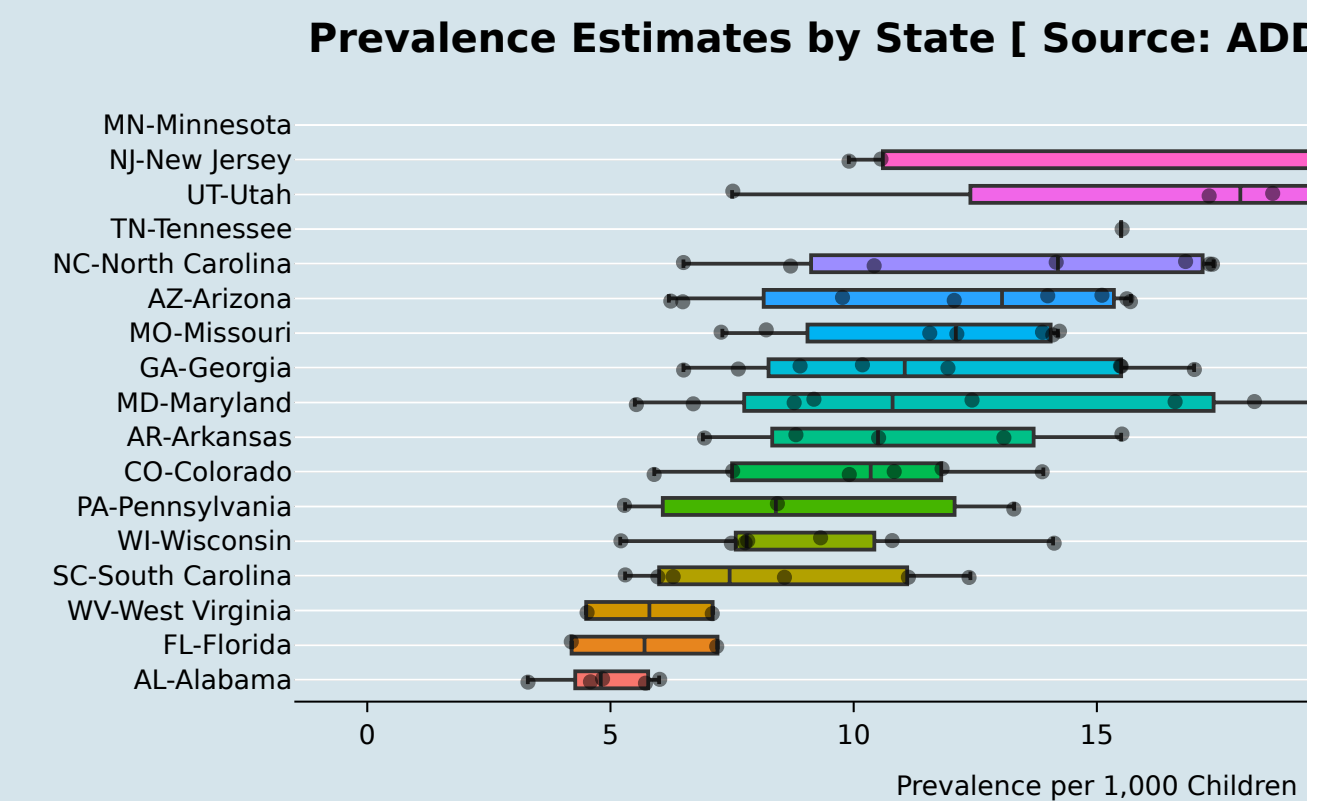
In [235]:
```
# Prevalence Estimates by State [ Source: ADDM ] , aggregated for different ye
p <- ggplot(ASD_State_ADDM, aes(x = reorder(State_Full2, Prevalence, FUN = med
                                y = Prevalence)) +
  geom_boxplot(aes(fill = reorder(State_Full2, Prevalence, FUN = median))) + #
  scale_fill_discrete(guide = guide_legend(title = "US. States")) + # Legend N
  #  geom_boxplot(fill = 'darkslategrey', alpha = 0.2) +
  scale_y_continuous(name = "Prevalence per 1,000 Children",
                     breaks = seq(0, 30, 5),
                     limits=c(0, 30)) +
  scale_x_discrete(name = "") +
  ggtitle("Prevalence Estimates by State [ Source: ADDM ]") +
  theme(title = element_text(face = 'bold.italic', color = "darkslategrey"),
        axis.title = element_text(face = 'plain', color = "darkslategrey"),
        legend.position = 'none') +
  coord_flip() + # Rotate chart
  geom_jitter(alpha = 0.5, position = position_jitter(width = 0.1)) # Add actu
```

In [236]:
```
# Show plot
p
```



In [237]:
```
# Theme of the economist magazine:
# p + theme_economist() + scale_colour_economist() + theme(legend.position = '
```

In [238]:
```
# Dynamic chart
p_dynamic <- p + theme_economist() + scale_colour_economist() + theme(legend.p
p_dynamic <- ggplotly(p_dynamic)
p_dynamic
```



Quiz:

Create Prevalence Estimates by State [Source: XXXX] for other three data sources:

In [239]:
```
# Write your code below and press Shift+Enter to execute
```

Double-click **here** for the solution.

Data Visualisation (Enhanced) - [R] US. State Level No. Children Surveyed by State [Source: ADDM] [Year 2014]

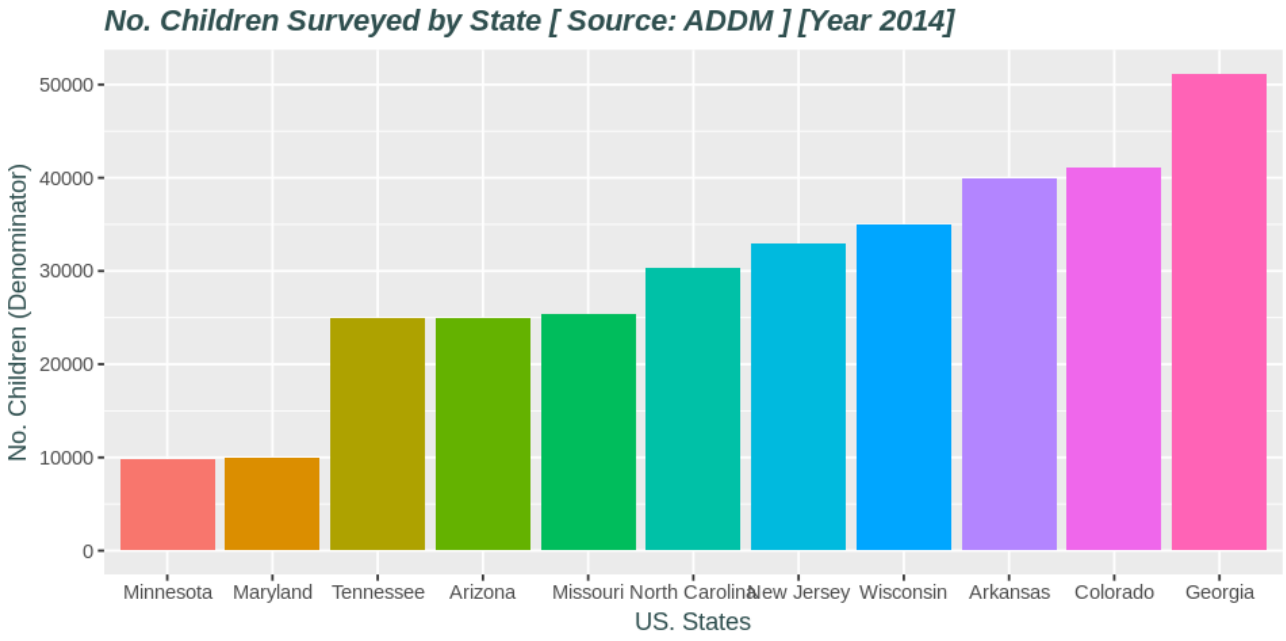
In [240]:
```
# Adjust in-line plot size to M x N
options(repr.plot.width=8, repr.plot.height=4)
```

Visualise: **No. Children Surveyed by State [Source: ADDM] [Year 2014]**

In [241]:
```
# All State Prevalence data with: Source == 'addm' & Year == 2014
# filter using dataframe: ASD_State_ADDM
ASD_State_Subset <- subset(ASD_State_ADDM, Year == 2014)
# or filer using dataframe: ASD_State
ASD_State_Subset <- subset(ASD_State, Source == 'addm' & Year == 2014)
```

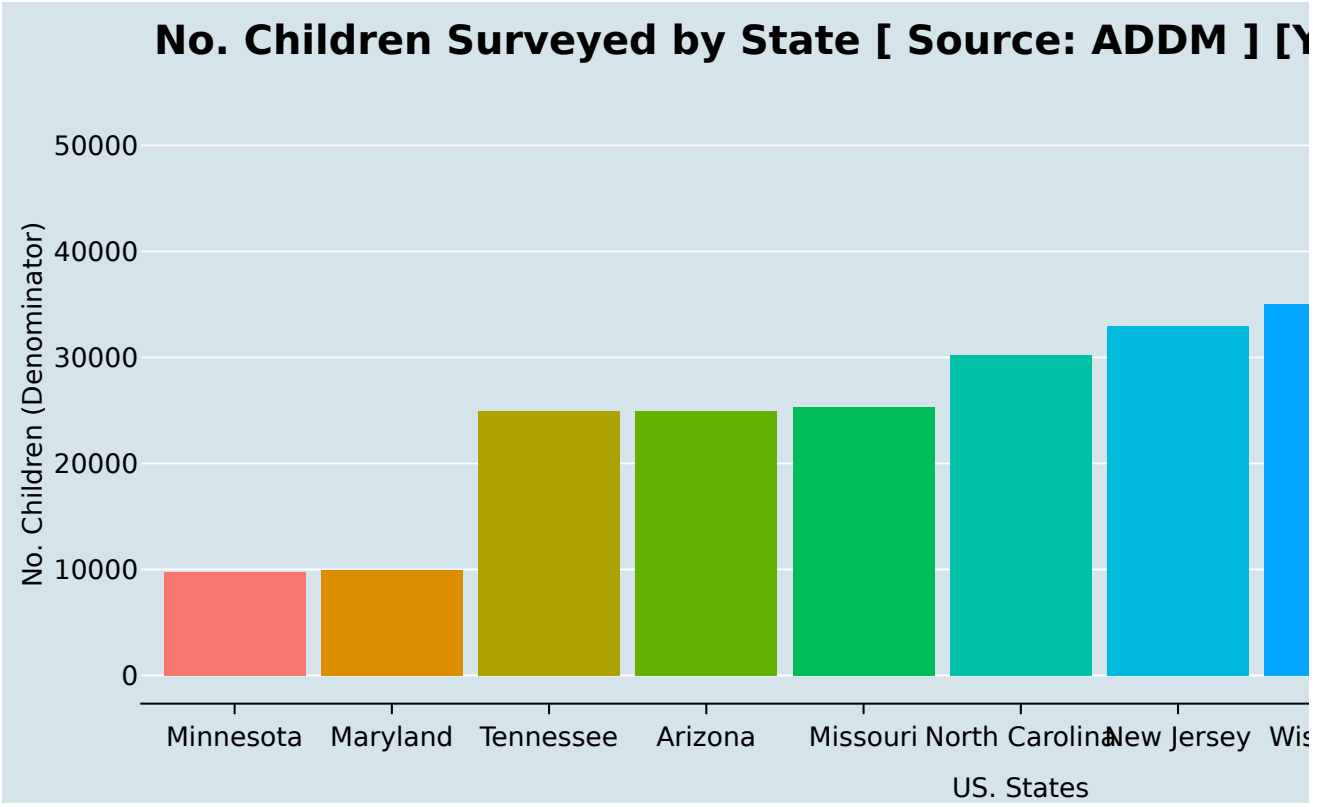
In [242]:
```
# Bar plot/chart for < No. Children surveyed by State [ADDM] [Year 2014] >
p <- ggplot(ASD_State_Subset, aes(x = reorder(State_Full1, Denominator, FUN =
                                  y = Denominator)) +
  geom_bar(stat="identity", aes(fill = reorder(State_Full1, Denominator, FUN =
  scale_fill_discrete(guide = guide_legend(title = "US. States")) + # Legend N
  scale_x_discrete(name = "US. States") +
  scale_y_continuous(name = "No. Children (Denominator)") +
  ggtitle("No. Children Surveyed by State [ Source: ADDM ] [Year 2014]") +
  #  geom_text(aes(label=Denominator), vjust=1.6, color="darkslategrey", size=
  theme(title = element_text(face = 'bold.italic', color = "darkslategrey"),
        axis.title = element_text(face = 'plain', color = "darkslategrey"),
        legend.position="none")
```

In [243]:
```
# Show plot
p
```



In [244]:
```
# Theme of the economist magazine:
# p + theme_economist() + scale_colour_economist() + theme(legend.position = '
```

In [245]:
```
# Dynamic chart
p_dynamic <- p + theme_economist() + scale_colour_economist() + theme(legend.p
p_dynamic <- ggplotly(p_dynamic)
p_dynamic
```



Quiz:

Create No. Children Surveyed by State [Source: XXXX] [Year CCYY] for other data sources & years:

In [246]:
```
# Write your code below and press Shift+Enter to execute
```

Double-click **here** for the solution.

Quiz:

Create No. ASD Children by State [Source: XXXX] [Year CCYY] for other data sources & years:

Hint: Use variable: ASD_State_ADDM$Numerator_ASD

In [247]:
```
# Write your code below and press Shift+Enter to execute
```

Double-click **here** for the solution.

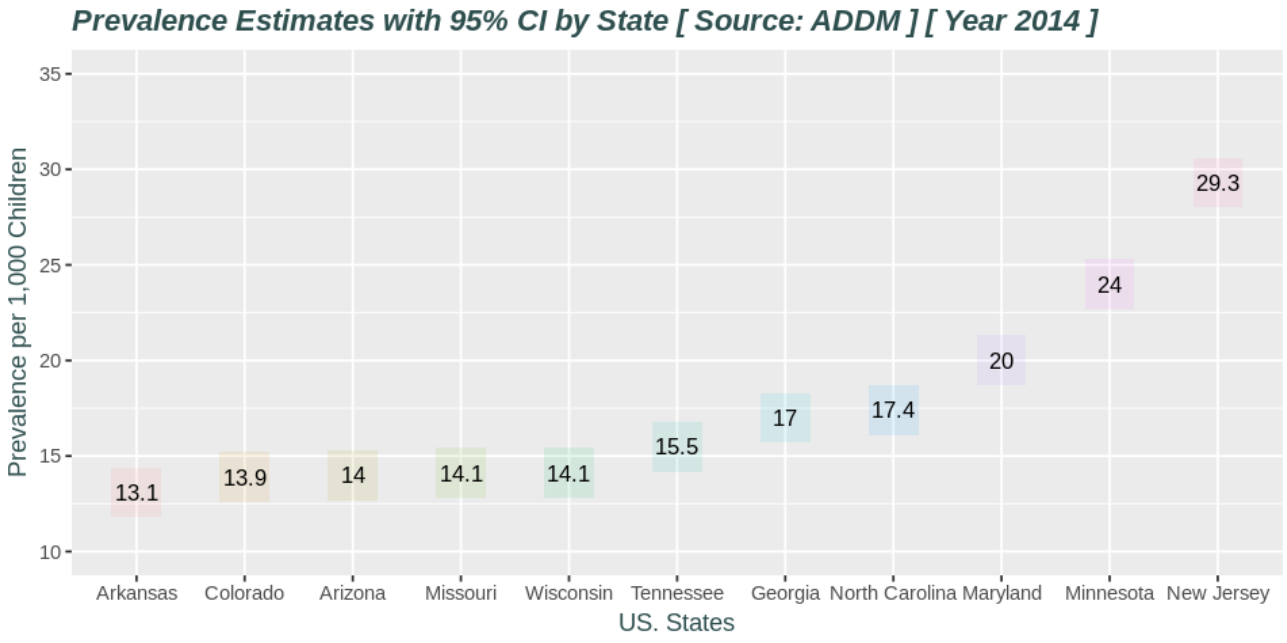
Data Visualisation (Enhanced) - [R] US. State Level Prevalence Estimates with 95% CI by State [Source: ADDM] [Year 2014]

In [248]:
```
# Adjust in-line plot size to M x N
# options(repr.plot.width=8, repr.plot.height=4)
```

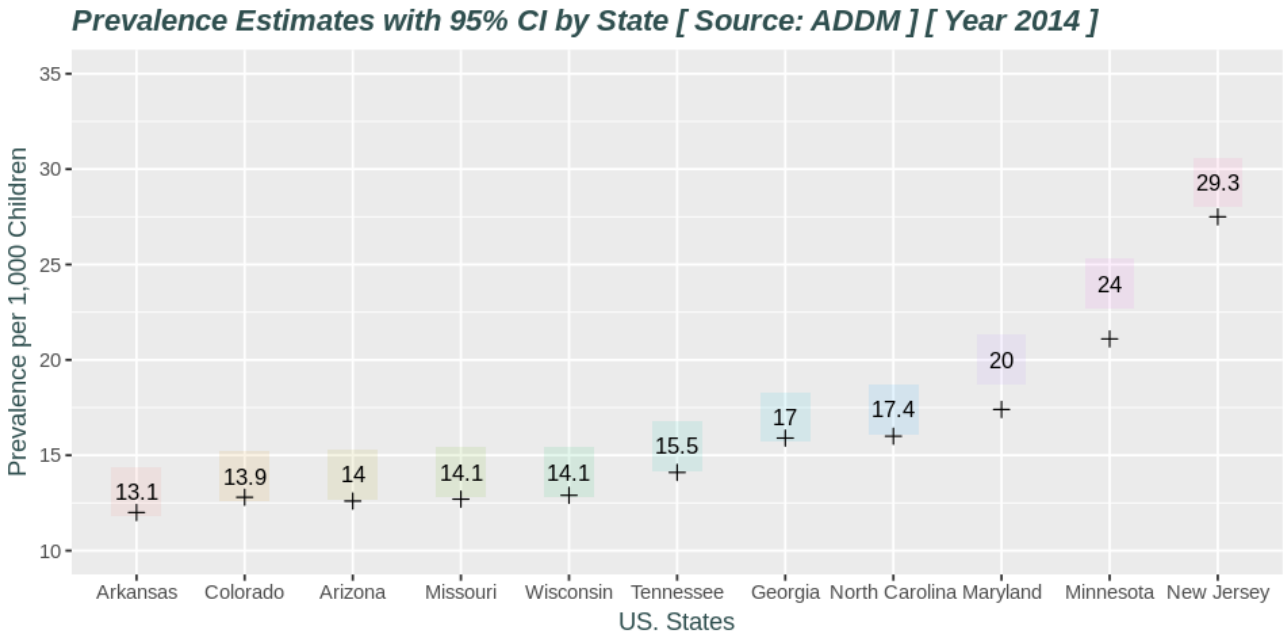
Visualise: **Prevalence Estimates with 95% CI by State [Source: ADDM] [Year 2014]**

In [249]:
```
# ASD_State_Subset <- subset(ASD_State_ADDM, Year == 2014)
# or
# ASD_State_Subset <- subset(ASD_State, Source == 'addm' & Year == 2014)

# Point plot/chart
p = ggplot(ASD_State_Subset, aes(x = reorder(State_Full1, Prevalence, median),
                                 y = Prevalence)) +
  geom_point(stat="identity", aes(colour = reorder(State_Full1, Prevalence, me
  scale_colour_discrete(guide = guide_legend(title = "US. States")) + # Legend
  scale_y_continuous(name = "Prevalence per 1,000 Children",
                     breaks = seq(10, 35, 5),
                     limits=c(10, 35)) +
  scale_x_discrete(name = "US. States") +
  ggtitle("Prevalence Estimates with 95% CI by State [ Source: ADDM ] [ Year 2
  theme(title = element_text(face = 'bold.italic', color = "darkslategrey"),
        axis.title = element_text(face = 'plain', color = "darkslategrey"),
        legend.position = 'none') +
  geom_text(aes(label=Prevalence), hjust=0.5, color="black", size=3.5)  # Show
```

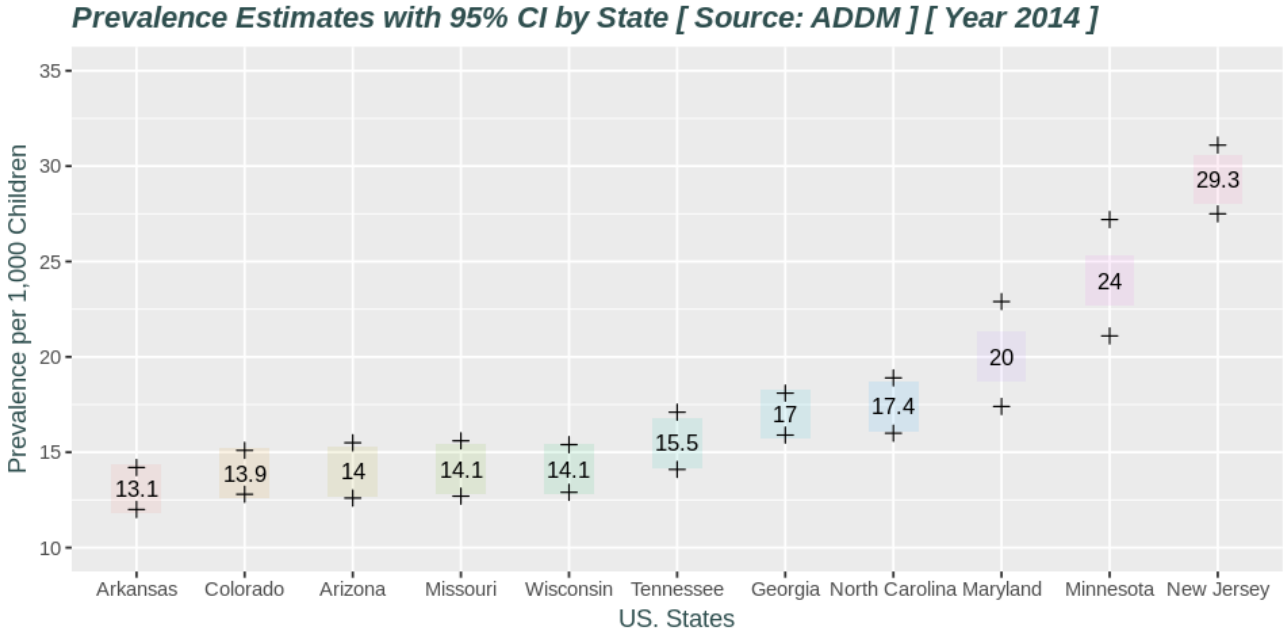
In [250]:
```
# Show plot
p
```



In [251]:
```
# Add Lower.CI
p = p + geom_point(data = ASD_State_Subset, aes(x = reorder(State_Full1, Preva
                                                shape=Source # point shape
),
size = 2 # point size
) +
  #  geom_text(aes(label=Lower.CI), hjust=-0.1, vjust=3, color="darkslategrey"
  scale_shape_manual(values=3)  # manual define point shape
# Show plot
p
```



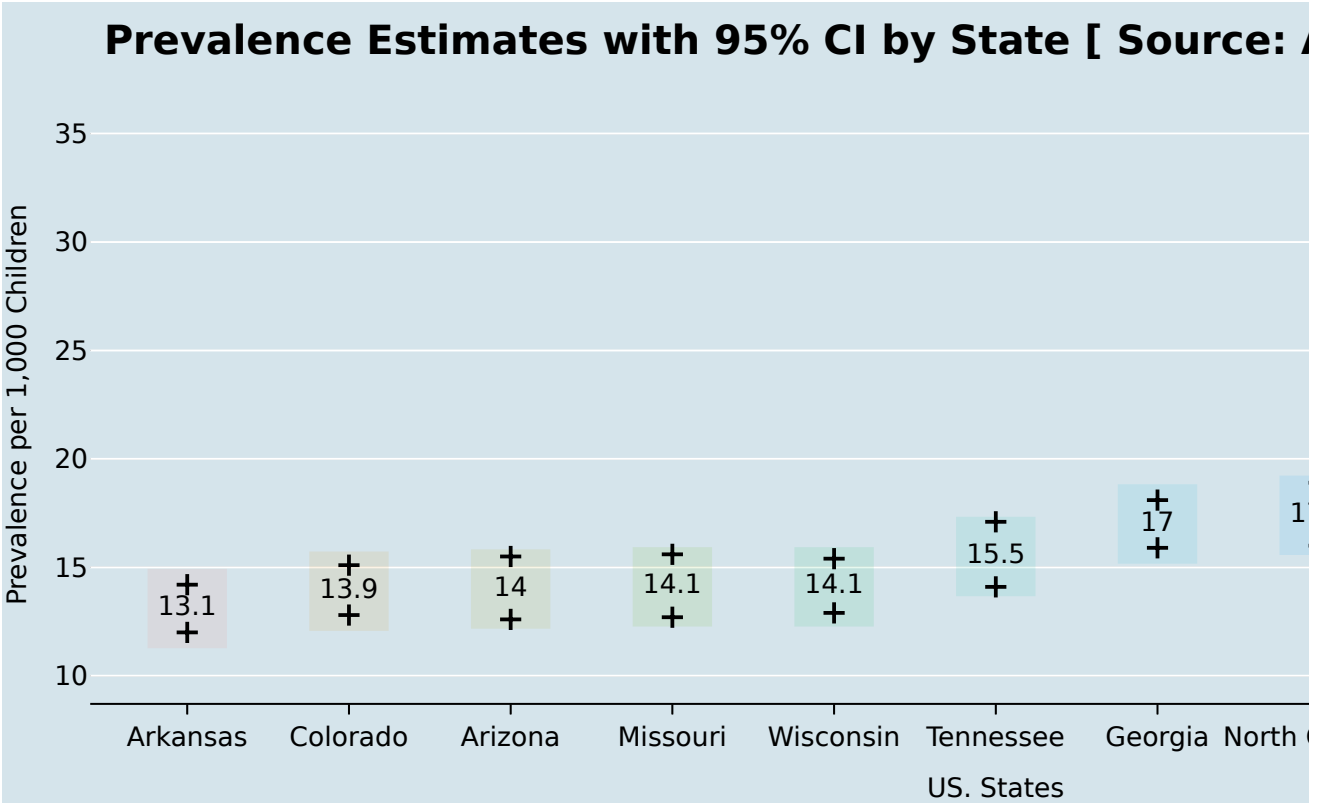
In [252]:
```
# Add Upper.CI
p = p + geom_point(data = ASD_State_Subset, aes(x = reorder(State_Full1, Preva
                                                shape=Source # point shape
),
size = 2 # point size
)
#  geom_text(aes(label=Upper.CI), hjust=-0.1, vjust=-3, color="darkslategrey",
# Show plot
p
```



In [253]:
```
# theme of the economist magazine:
# p + theme_economist() + scale_colour_economist() + scale_colour_discrete(gui
```

In [254]:
```
# Dynamic chart
p_dynamic <- p + theme_economist() + scale_colour_economist() + scale_colour_d
p_dynamic <- ggplotly(p_dynamic)
p_dynamic
```

```
Scale for 'colour' is already present. Adding another scale for 'colour',
which will replace the existing scale.
Scale for 'colour' is already present. Adding another scale for 'colour',
which will replace the existing scale.
```



Quiz:

Create Prevalence Estimates with 95% CI by State [Source: ADDM] [Year CCYY] for other data sources & years:

In [255]:
```
# Write your code below and press Shift+Enter to execute
```

Double-click **here** for the solution.

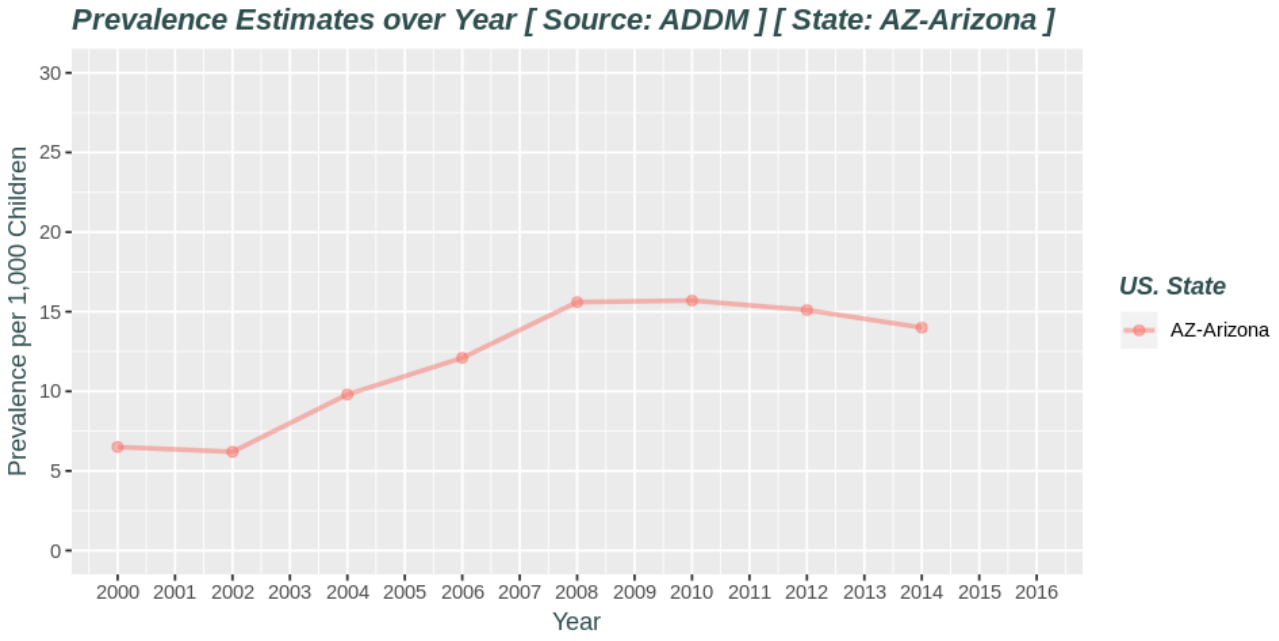
Data Visualisation (Enhanced) - [R] US. State Level Prevalence Estimates over Year [Source: ADDM] [State: AZ-Arizona]

In [256]:
```
# Adjust in-line plot size to M x N
# options(repr.plot.width=8, repr.plot.height=4)
```

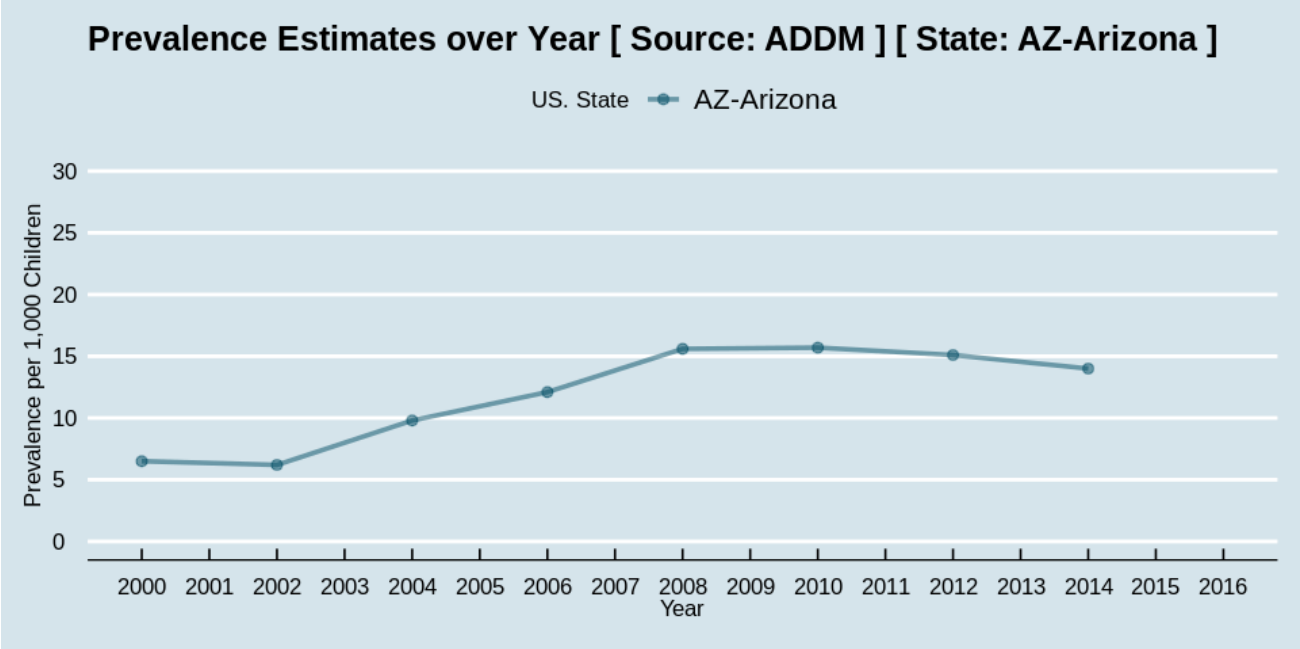
Visualise: **Prevalence Estimates over Year [Source: ADDM] [State: AZ-Arizona]**

In [257]:
```
# All year/time Prevalence data with: Source_UC == 'ADDM' & State_Full2 == 'AZ
ASD_State_Subset <- subset(ASD_State, Source_UC == 'ADDM' & State_Full2 == 'AZ

# Line plot/chart for < State ASD Prevalence [ADDM] [AZ-Arizona] >
p <- ggplot(ASD_State_Subset, aes(x = Year, y = Prevalence))
# Select (add) line chart type:
p <- p + geom_line(aes(color = State_Full2),
                   linetype = "solid",  # http://sape.inf.usi.ch/quick-referen
                   size=1,
                   alpha=0.5)
# Select (add) points to chart:
p <- p + geom_point(aes(color = State_Full2),
                    size=3,
                    shape=20,
                    alpha=0.5)
# Customize legend name:
p <- p + labs(color = "US. State")
# Adjust x and y axis, scale, limit and labels:
p <- p + scale_y_continuous(name = "Prevalence per 1,000 Children",
                            breaks = seq(0, 30, 5),
                            limits=c(0, 30)) +
  scale_x_continuous(name = "Year",
                     breaks = seq(2000, 2016, 1),
                     limits = c(2000, 2016))
# Customize chart title:
p <- p + ggtitle("Prevalence Estimates over Year [ Source: ADDM ] [ State: AZ-
# Customize chart title and axis labels:
p <- p + theme(title = element_text(face = 'bold.italic', color = "darkslategr
               axis.title = element_text(face = 'plain', color = "darkslategre
```

In [258]:
```
# Show plot
p
```



In [259]:
```
# Theme of the economist magazine:
p + theme_economist() + scale_colour_economist()
```



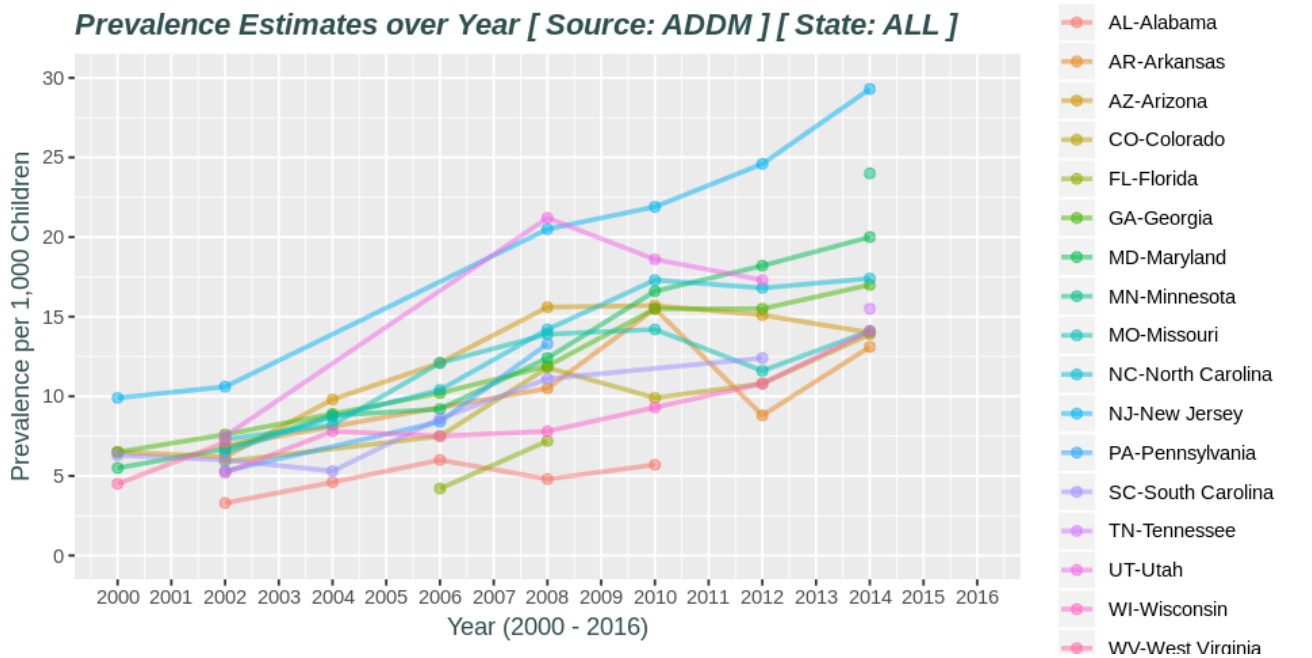
Data Visualisation (Enhanced) - [R] US. State Level Prevalence Estimates over Year [Source: ADDM] [State: ALL]

In [260]:
```
# Adjust in-line plot size to M x N
# options(repr.plot.width=8, repr.plot.height=4)
```

Visualise: **Prevalence Estimates over Year [Source: ADDM] [State: ALL]**

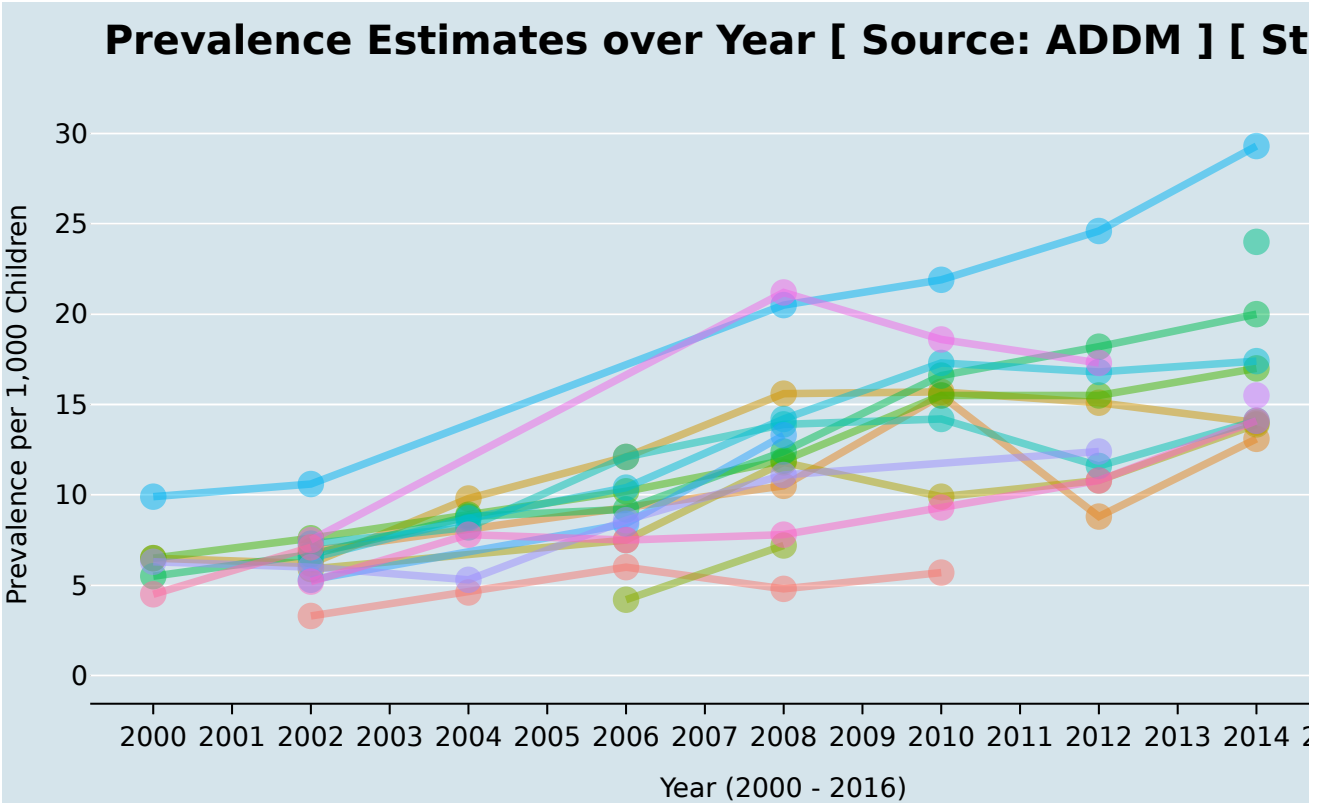
In [261]:
```
p <- ggplot(ASD_State_ADDM, aes(x = Year, y = Prevalence))
# Select (add) line chart type:
p <- p + geom_line(aes(color = State_Full2),
                   linetype = "solid",  # http://sape.inf.usi.ch/quick-referen
                   size=1,
                   alpha=0.5)
# Select (add) points to chart:
p <- p + geom_point(aes(color = State_Full2),
                    size=3,
                    shape=20,
                    alpha=0.5)
# Show plot
# p
# Customize line color and legend name:
p <- p + labs(color = "US. State")
# Adjust x and y axis, scale, limit and labels:
p <- p + scale_y_continuous(name = "Prevalence per 1,000 Children",
                            breaks = seq(0, 30, 5),
                            limits=c(0, 30)) +
  scale_x_continuous(name = "Year (2000 - 2016)",
                     breaks = seq(2000, 2016, 1),
                     limits = c(2000, 2016))
# Customize chart title:
p <- p + ggtitle("Prevalence Estimates over Year [ Source: ADDM ] [ State: ALL
# Customize chart title and axis labels:
p <- p + theme(title = element_text(face = 'bold.italic', color = "darkslategr
               axis.title = element_text(face = 'plain', color = "darkslategre
               legend.position="right")
```

In [262]:
```
# Show plot
p
```



In [263]:
```
# Dynamic chart
p_dynamic <- p + theme_economist() + scale_colour_economist() + scale_colour_d
p_dynamic <- ggplotly(p_dynamic)
p_dynamic
```

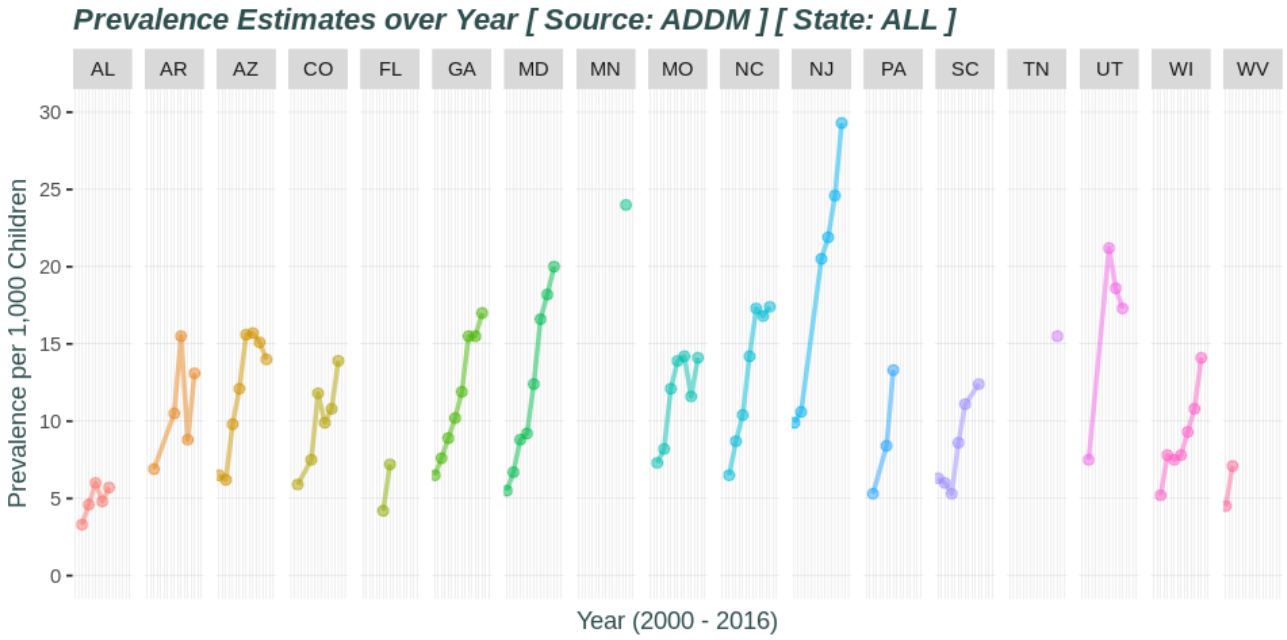
```
Scale for 'colour' is already present. Adding another scale for 'colour',
which will replace the existing scale.
```



Split chart by state

In [264]:
```
# Show plot in facet_grid
p + facet_grid(facets = . ~ State) +
  theme(legend.position = "none", # Hide legend
        axis.text.x=element_blank(),  # Hide axis
        axis.ticks.x=element_blank(), # Hide axis
        panel.background = element_blank(), # Remove panel background
        panel.grid.major = element_line(size = 0.1, linetype = 1, colour = "li
  )
```

```
geom_path: Each group consists of only one observation. Do you need to adjust
the group aesthetic?
geom_path: Each group consists of only one observation. Do you need to adjust
the group aesthetic?
```

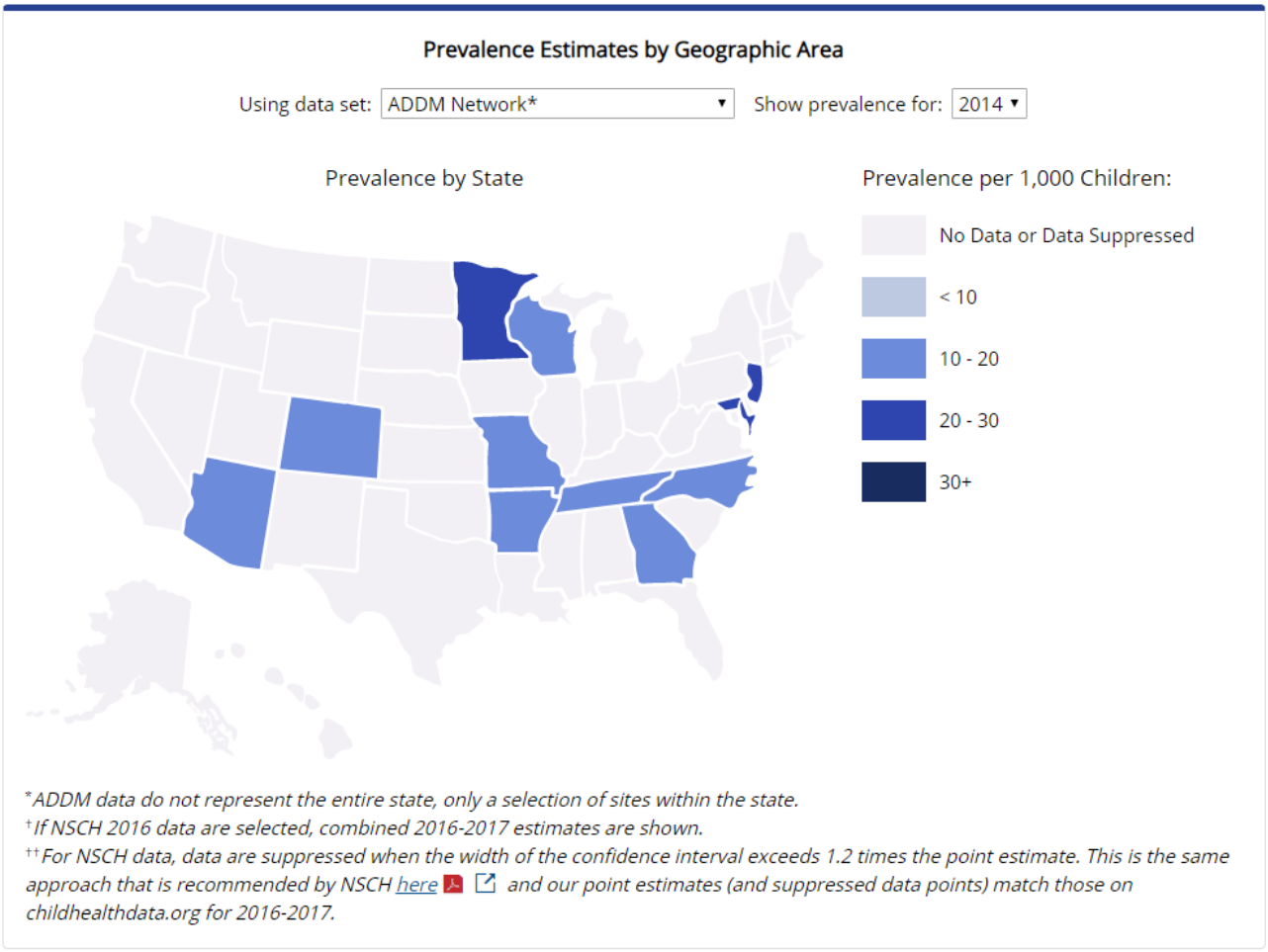


Data Visualisation (Enhanced) - Plotting on Map

In [265]:
```
# ---------------------------------
# EDA - Visualisation on map
# ---------------------------------
if(!require(usmap)){install.packages("usmap")}
library(usmap) # usmap: Mapping the US
```

```
Loading required package: usmap
```

Data Visualisation (Enhanced) - Plotting on Map [CDC] REPORTED PREVALENCE VARIES BY GEOGRAPHIC LOCATION

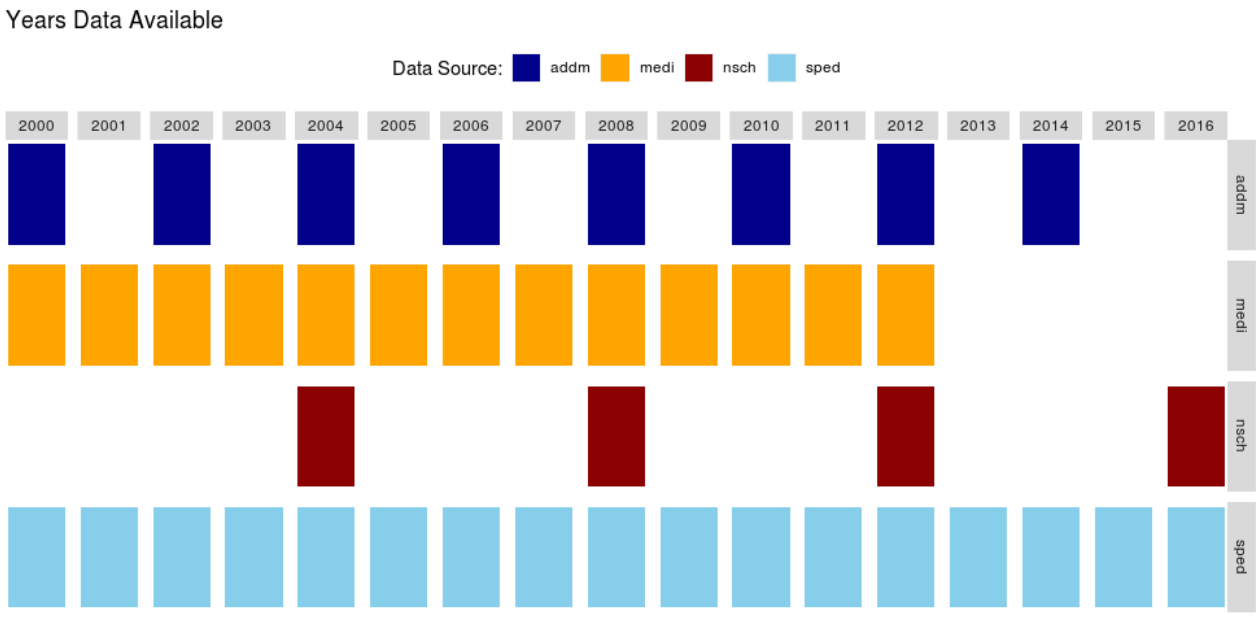


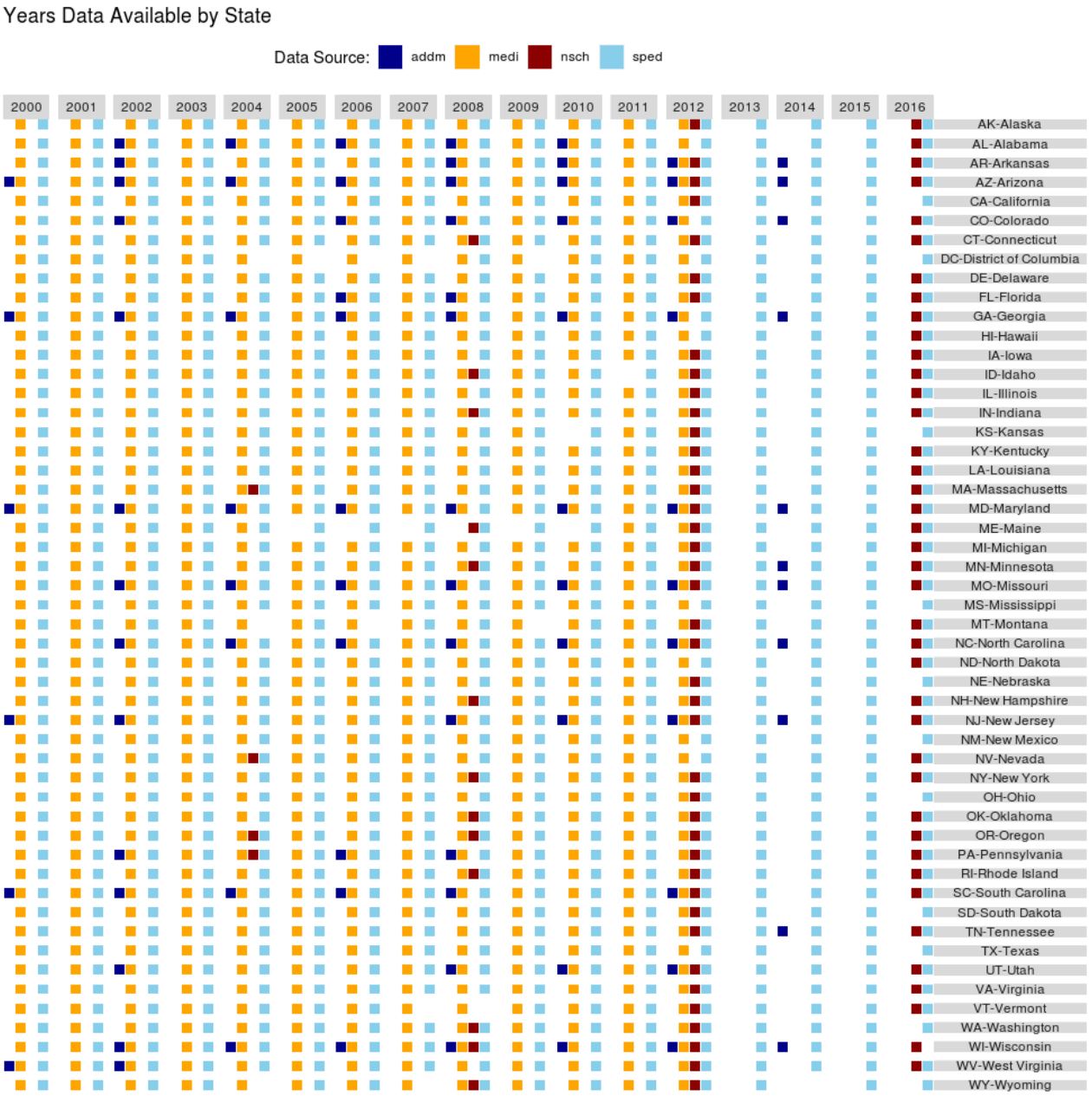
*ADDM data do not represent the entire state, only a selection of sites within the state.
†If NSCH 2016 data are selected, combined 2016-2017 estimates are shown.
††For NSCH data, data are suppressed when the width of the confidence interval exceeds 1.2 times the point estimate. This is the same approach that is recommended by NSCH here and our point estimates (and suppressed data points) match those on childhealthdata.org for 2016-2017.

Data Visualisation (Enhanced) - Plotting on Map [R] REPORTED PREVALENCE VARIES BY GEOGRAPHIC LOCATION

Let's review data availability by data Sources & Years:

- ASD_State_ADDM in Years: 2000, 2002, 2004, 2006, 2008, 2010, 2012, 2014
- ASD_State_MEDI in Years: 2000 ~ 2012
- ASD_State_NSCH in Years: 2004, 2008, 2012, 2016
- ASD_State_SPED in Years: 2000 ~ 2016





Data Visualisation (Enhanced) - Plotting on Map [R] REPORTED PREVALENCE VARIES BY GEOGRAPHIC LOCATION [Source: ADDM] [Year: 2014]

In [266]:
```
# Adjust in-line plot size to M x N
# options(repr.plot.width=8, repr.plot.height=4)
```

Prepare US State level data: [Source: ADDM] [Year: 2014]

In [267]:
```
# Prepare data - addm 2014
Map_Data_Source = 'addm' # Available values lowercase: 'addm', 'medi', 'nsch',
Map_Data_Value = 'Prevalence' # variable must be numeric, variable name in 'qu

# Uncomment below to use Prevalence of different groups:
# Map_Data_Value = 'Male.Prevalence' # variable must be numeric, variable name
# Map_Data_Value = 'Female.Prevalence' # variable must be numeric, variable na
# Map_Data_Value = 'Asian.or.Pacific.Islander.Prevalence' # variable must be n

Map_Data_Year = 2014 # must be integer
ASD_State_Subset = subset(ASD_State, Source == Map_Data_Source & Year == Map_D
```

The usmap package/function requires input data to have a column of state, or fips. (case sensitive)

- state: Name of US state
- fips: FIPS code for either a US state

https://cran.r-project.org/web/packages/usmap/vignettes/mapping.html (https://cran.r-project.org/web/packages/usmap/vignettes/mapping.html)

https://cran.r-project.org/web/packages/usmap/usmap.pdf (https://cran.r-project.org/web/packages/usmap/usmap.pdf)

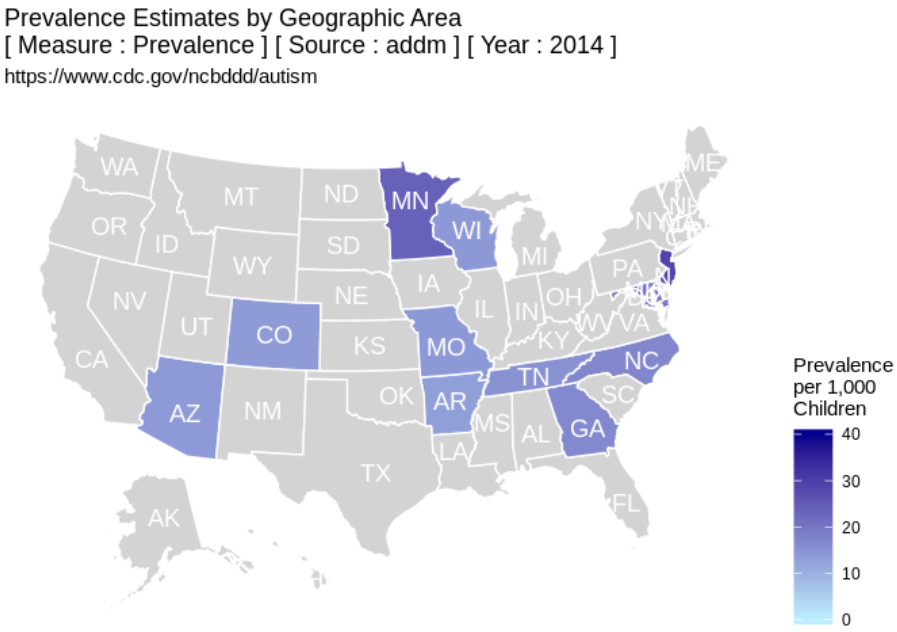
In [268]:
```
# The usmap package/function requires input data to have a column of 'state',
ASD_State_Subset$state = ASD_State_Subset$State
# Glance
head(ASD_State_Subset)
```

State	Denominator	Prevalence	Lower.CI	Upper.CI	Year	Source	Source_Full1	State_Full1	State_Full
AZ	24952	14.0	12.6	15.5	2014	addm	Autism & Developmental Disabilities Monitoring Network	Arizona	AZ-Arizona
AR	39992	13.1	12.0	14.2	2014	addm	Autism & Developmental Disabilities Monitoring Network	Arkansas	AR Arkansa
CO	41128	13.9	12.8	15.1	2014	addm	Autism & Developmental Disabilities Monitoring Network	Colorado	CO Colorad
GA	51161	17.0	15.9	18.1	2014	addm	Autism & Developmental Disabilities Monitoring Network	Georgia	GA-Georgi
MD	9955	20.0	17.4	22.9	2014	addm	Autism & Developmental Disabilities Monitoring Network	Maryland	MD Marylan
MN	9767	24.0	21.1	27.2	2014	addm	Autism & Developmental Disabilities Monitoring Network	Minnesota	MN Minnesot

Visualise: **Prevalence Estimates by Geographic Area** [Source: ADDM] [Year: 2014]

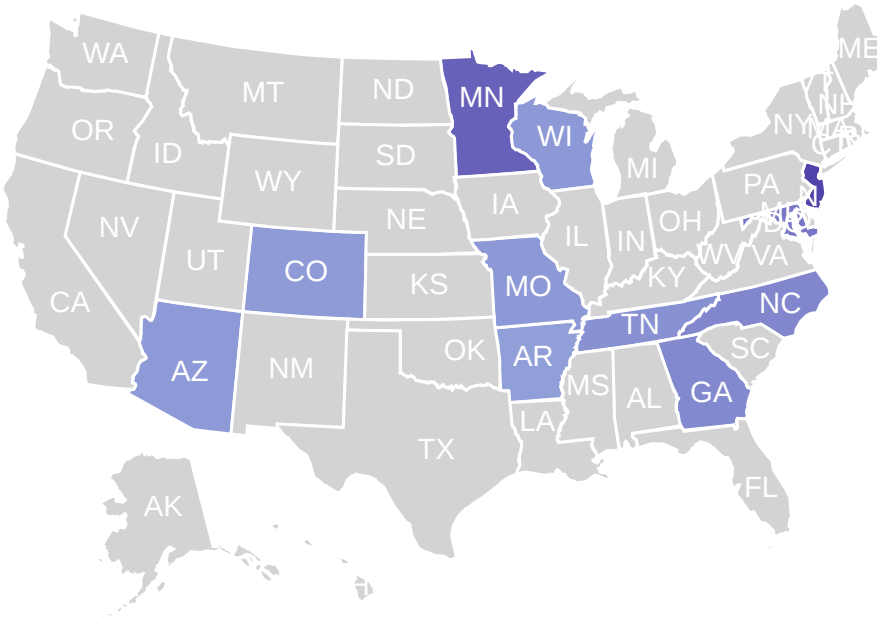
In [269]:

```
# Show data on map
p_map_addm_2014 <- plot_usmap(data = ASD_State_Subset, values = Map_Data_Value
                              color = "white", # map line colour
                              labels = TRUE,  # State name shown
                              label_color = 'white' # State name colour
) +
  scale_fill_continuous(
    na.value = "lightgrey", # Set colour with no State data
    low="lightblue1", high = "darkblue",
    name = "Prevalence\nper 1,000\nChildren",
    limits=c(0, 40) #same colour levels/limits for plots
  ) +
  labs(title = paste("Prevalence Estimates by Geographic Area", '\n[ Measure :
       subtitle = 'https://www.cdc.gov/ncbddd/autism'
  ) +
  theme(panel.background = element_rect(color = "white", fill = "white"),
        legend.position = "right")
# Show map
p_map_addm_2014
```



In [270]:
```
# Dynamic map
p_dynamic <- p_map_addm_2014
p_dynamic <- ggplotly(p_dynamic)
p_dynamic
```

Prevalence Estimates by Geographic Area
[Measure : Prevalence] [Source : addm] [Year : 2014]



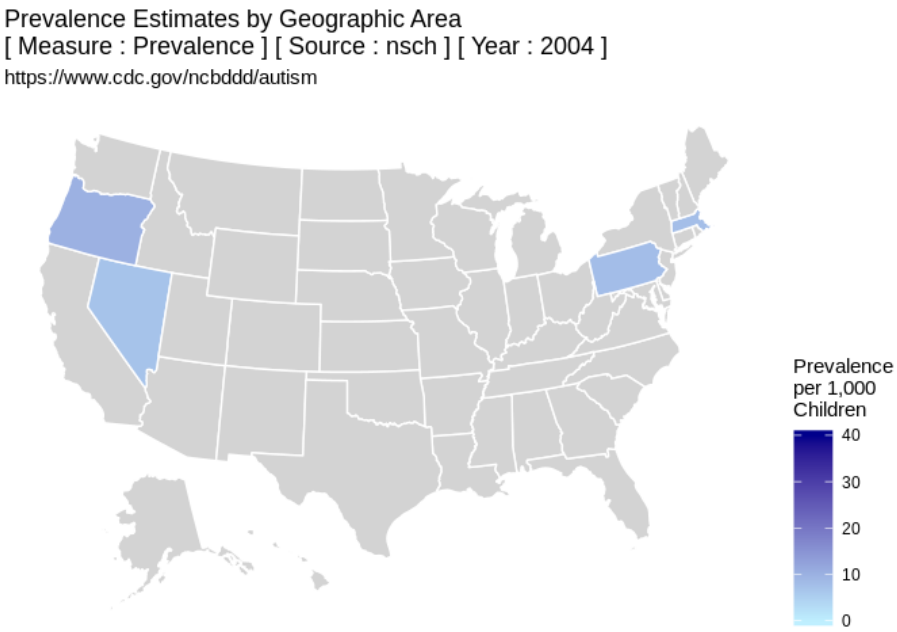
Data Visualisation (Enhanced) - Plotting on Map [R] REPORTED PREVALENCE VARIES BY GEOGRAPHIC LOCATION [Source: NSCH] [Year: 2004, 2008, 2012, 2016]

Prepare US State level data: [Source: NSCH] [Year: ALL]

In [271]:
```
Map_Data_Source = 'nsch' # Available values lowercase: 'addm', 'medi', 'nsch',
Map_Data_Value = 'Prevalence' # variable must be numeric, variable name in 'qu
```

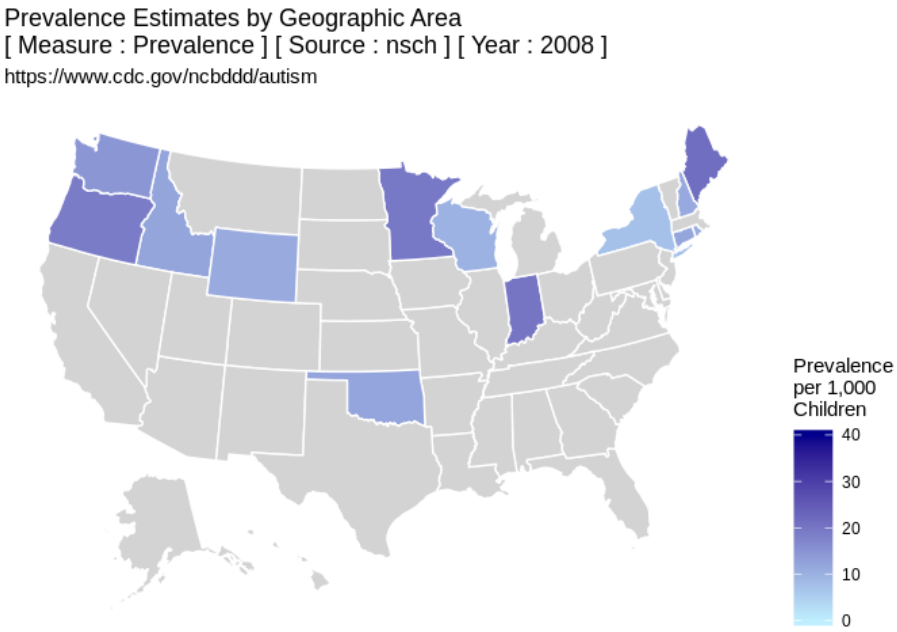
Visualise: **Prevalence Estimates by Geographic Area** [Source: NSCH] [Year: 2004]

In [272]:
```
# Prepare data - nsch 2004
Map_Data_Year = 2004 # must be integer
ASD_State_Subset = subset(ASD_State, Source == Map_Data_Source & Year == Map_D
ASD_State_Subset$state = ASD_State_Subset$State
# Plot on map
p_map_nsch_2004 <- plot_usmap(data = ASD_State_Subset, values = Map_Data_Value
p_map_nsch_2004
```



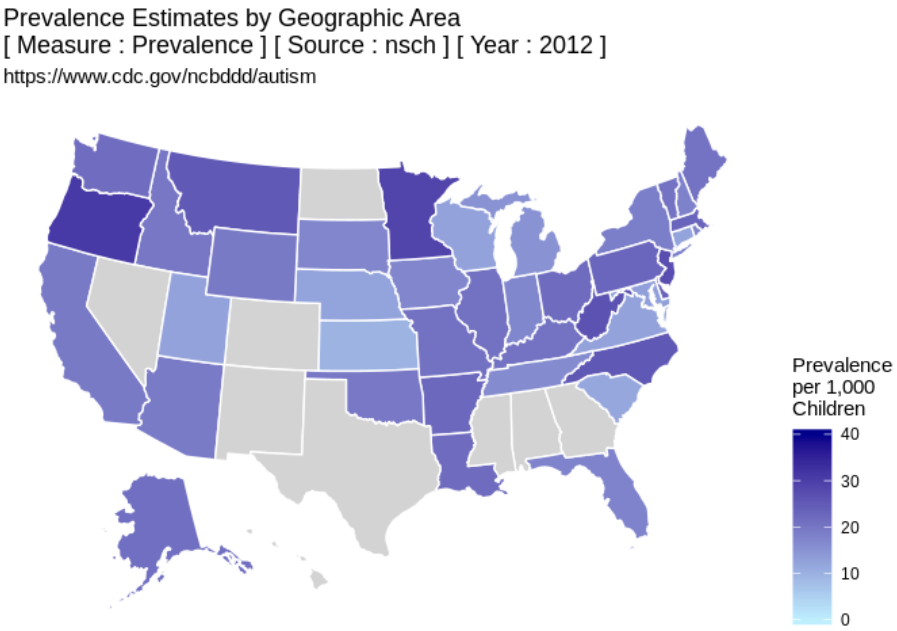
Visualise: **Prevalence Estimates by Geographic Area** [Source: NSCH] [Year: 2008]

In [273]:
```
# Prepare data - nsch 2008
Map_Data_Year = 2008 # must be integer
ASD_State_Subset = subset(ASD_State, Source == Map_Data_Source & Year == Map_D
ASD_State_Subset$state = ASD_State_Subset$State
p_map_nsch_2008 <- plot_usmap(data = ASD_State_Subset, values = Map_Data_Value
p_map_nsch_2008
```



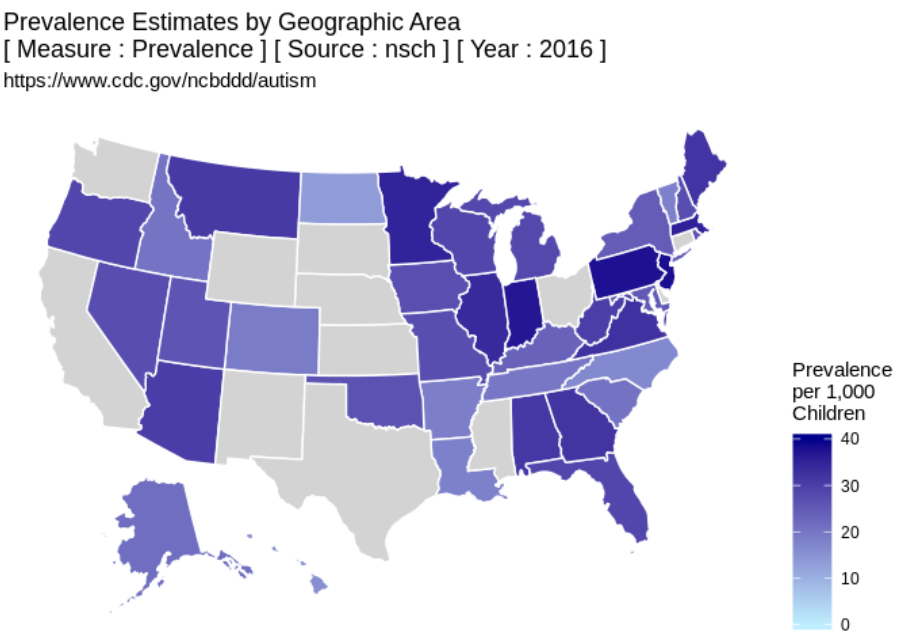
Visualise: **Prevalence Estimates by Geographic Area** [Source: NSCH] [Year: 2012]

In [274]:
```
# Prepare data - nsch 2012
Map_Data_Year = 2012 # must be integer
ASD_State_Subset = subset(ASD_State, Source == Map_Data_Source & Year == Map_D
ASD_State_Subset$state = ASD_State_Subset$State
p_map_nsch_2012 <- plot_usmap(data = ASD_State_Subset, values = Map_Data_Value
p_map_nsch_2012
```



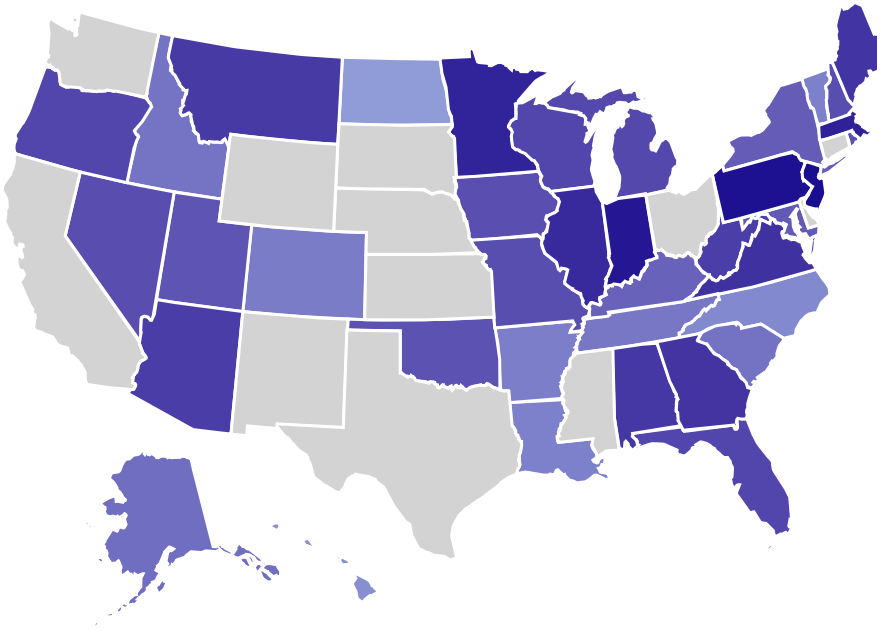
Visualise: **Prevalence Estimates by Geographic Area** [Source: NSCH] [Year: 2016]

In [275]:
```
# Prepare data - nsch 2016
Map_Data_Year = 2016 # must be integer
ASD_State_Subset = subset(ASD_State, Source == Map_Data_Source & Year == Map_D
ASD_State_Subset$state = ASD_State_Subset$State
p_map_nsch_2016 <- plot_usmap(data = ASD_State_Subset, values = Map_Data_Value
p_map_nsch_2016
```



In [276]:
```
# Dynamic map
p_dynamic <- p_map_nsch_2016 # [ Source: NSCH ] [ Year: 2016 ]
p_dynamic <- ggplotly(p_dynamic)
p_dynamic
```

Prevalence Estimates by Geographic Area
[Measure : Prevalence] [Source : nsch] [Year : 2016]

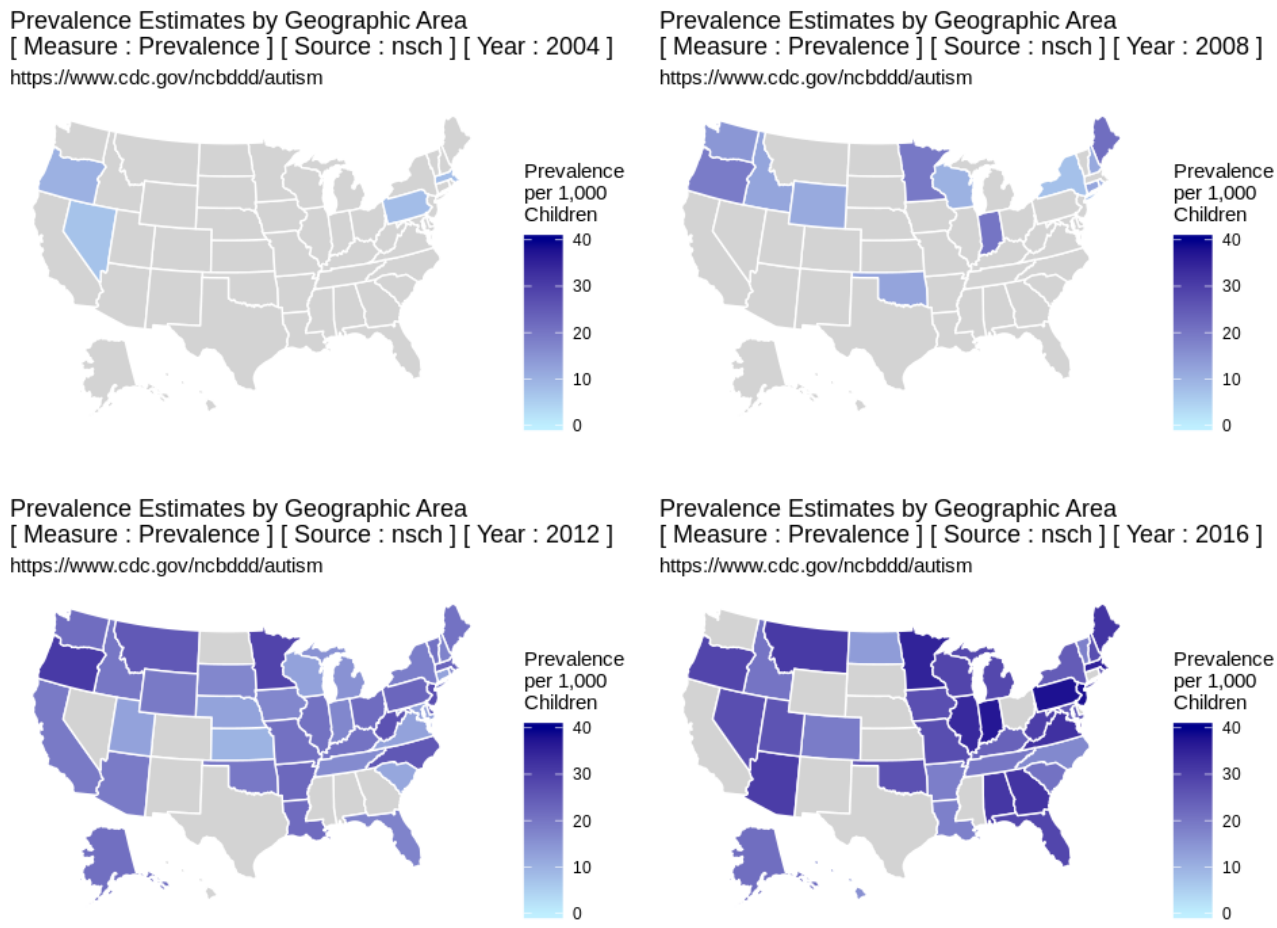


Combine multiple plots to show in one page/screen:

In [277]:
```
# Adjust in-line plot size to M x N
options(repr.plot.width=8, repr.plot.height=6)
```

In [278]:
```
# ---------------------------------
# Combine multiple plots
# ---------------------------------
if(!require(cowplot)){install.packages("cowplot")}
library('cowplot')
cowplot::plot_grid(
  p_map_nsch_2004,
  p_map_nsch_2008,
  p_map_nsch_2012,
  p_map_nsch_2016,
  nrow = 2)
```

```
Loading required package: cowplot

************************************************************
Note: As of version 1.0.0, cowplot does not change the
  default ggplot2 theme anymore. To recover the previous
  behavior, execute:
  theme_set(theme_cowplot())
************************************************************


Attaching package: ‘cowplot’

The following object is masked from ‘package:ggthemes’:

    theme_map
```



Export current plot as image file:

In [279]:
```
# ---------------------------------
# Export current plot as image file
# ---------------------------------
ggsave("plot Map Prevalence Estimates by Geographic Area [NSCH] [2004-2016].pn
       width = 60, height = 30, units = 'cm')
```

()

Sampling & Normality

Sampling & Normality - Population & Sample

In [280]:
```
# Adjust in-line plot size to M x N
# options(repr.plot.width=8, repr.plot.height=4)
```

Create a Population of US. State level ASD Prevalence from Source SPED in Year 2016

In [281]:
```
# ---------------------------------
# Create a *Population* of US. State level ASD Prevalence from Source SPED in
# ---------------------------------
ASD_State_SPED_2016 <- subset(ASD_State, Source == 'sped' & Year == 2016, sele
#
head(ASD_State_SPED_2016)
```

State	Prevalence
AL	9.1
AK	10.1
AZ	10.4
AR	9.5
CA	13.9
CO	7.3

In [282]:
```
dim(ASD_State_SPED_2016)
# *Population* mean Prevalence
mean(ASD_State_SPED_2016$Prevalence)
```

50 2

11.182

Define a function to calculate population std-dev (Omega):

In [283]:
```
# Use sd()   to calculate *sample*     std-dev (S)
# Use sd.p() to calculate *population* std-dev (Omega)

# Define a function sd.p() to calculate *population* std-dev (Omega)
# https://www.dummies.com/education/math/statistics/standard-deviation-r/

sd.p = function(x) {sd(x) * sqrt((length(x)-1)/length(x))}

# Treat as sample:
cat('sd()   of ASD_State_SPED_2016$Prevalence : ', sd(ASD_State_SPED_2016$Prev

# Treat as population:
cat('\nsd.p() of ASD_State_SPED_2016$Prevalence : ', sd.p(ASD_State_SPED_2016$
```

```
sd()   of ASD_State_SPED_2016$Prevalence :  3.233226
sd.p() of ASD_State_SPED_2016$Prevalence :  3.200731
```

Sampling & Normality - Central Limit Theorem (CLT)

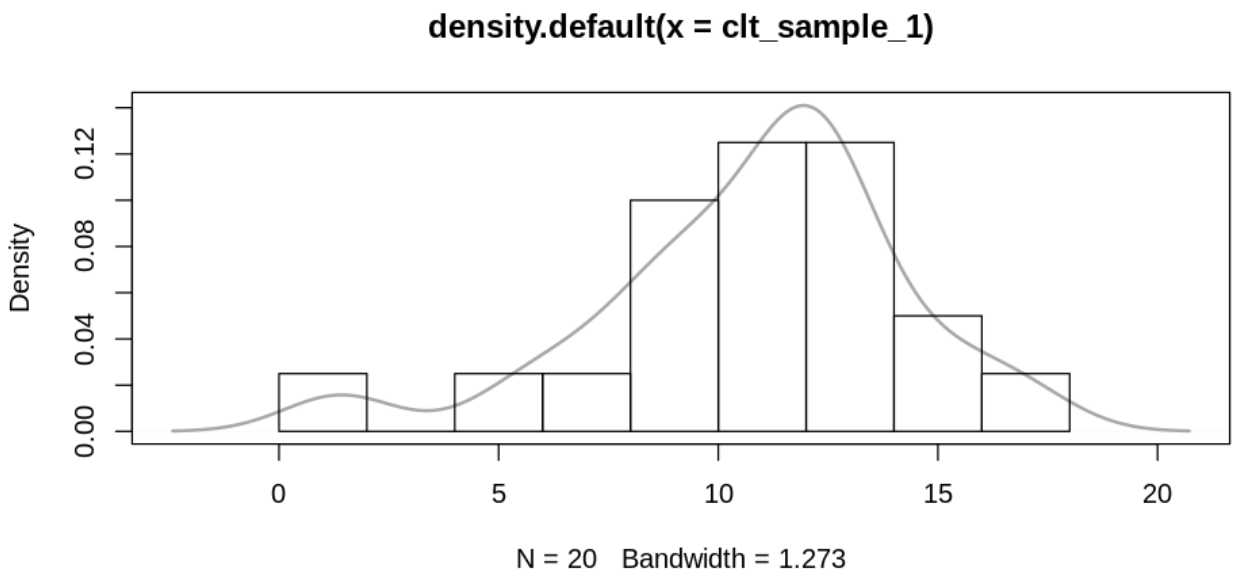
Create a Sample of US. State level ASD Prevalence from Source SPED in Year 2016

In [284]:
```
# Create a *Sample* from ASD_State_SPED_2016$Prevalence,
# with sample size n =
clt_n = 20
# clt_n = 40

set.seed(88)
clt_sample_1 = sample(x = ASD_State_SPED_2016$Prevalence, size = clt_n, replac
clt_sample_1
```

```
11.2  9.5  16.9  6.9  11.2  8.2  12.7  9.5  1.4  12.7  10.1  14.2  11.9  11.9  8.5  12.1  12.1
15.4  13  5.5
```

In [285]:
```
# Adjust in-line plot size to M x N
options(repr.plot.width=8, repr.plot.height=4)
```

In [286]:
```
plot(density(clt_sample_1), col="darkgrey", lwd=2)
hist(clt_sample_1, probability = T, add = T)
```



Draw a large k number of samples, with sample size = n:

In [287]:
```
# Repeatedly sample for k times, create a matrix/array to store these samples
clt_k = 10000 # or called 'N', but this can be confusing due to N can also be

set.seed(88) # Repeatable sampling using pseudo random method
clt_sample_k <- (replicate(clt_k, sample(x = ASD_State_SPED_2016$Prevalence, s

# first few samples
clt_sample_k[, 1:6]
```

11.2	9.5	11.2	8.3	8.3	6.9
9.5	11.9	10.4	11.9	11.9	15.2
16.9	6.9	9.3	1.4	15.4	19.4
6.9	13.0	19.4	12.7	8.5	15.1
11.2	8.3	12.1	9.1	11.2	12.1
8.2	12.1	10.1	14.2	10.2	1.4
12.7	10.8	12.1	8.6	9.5	15.1
9.5	15.2	14.1	14.1	10.8	6.9
1.4	8.7	12.5	16.7	9.8	9.5
12.7	9.0	5.5	13.9	19.4	14.2
10.1	9.3	9.5	9.5	14.2	19.4
14.2	12.1	16.7	12.5	14.1	16.9
11.9	1.4	9.6	10.8	11.9	9.6
11.9	11.9	10.2	10.3	5.5	14.1
8.5	10.8	15.2	6.9	10.3	14.2
12.1	14.1	15.1	1.4	15.4	19.4
12.1	15.4	16.9	14.1	14.2	8.6
15.4	8.5	12.1	14.1	13.0	9.5
13.0	1.4	19.4	12.1	15.4	9.0
5.5	8.5	12.7	10.8	11.9	19.4

In [288]:
```
# last sample
clt_sample_k[, clt_k]
```

8.6 14.1 12.1 19.4 12.7 10.3 10.3 11.2 10.4 9 10.4 13.9 9.3 14.2 11 12.1 9.5
10.1 13 15.2

In [289]:
```
# mean values of first few samples
mean(clt_sample_k[, 1])
mean(clt_sample_k[, 2])
mean(clt_sample_k[, 3])
mean(clt_sample_k[, 4])
mean(clt_sample_k[, 5])
mean(clt_sample_k[, 6])

# or use apply() function to loop
apply(clt_sample_k[, 1:6], 2, mean)
```

10.745

9.94

12.705

10.67

12.045

12.795

10.745 9.94 12.705 10.67 12.045 12.795

In [290]:
```
# std-dev values of first few samples
sd(clt_sample_k[, 1])
sd(clt_sample_k[, 2])
sd(clt_sample_k[, 3])
sd(clt_sample_k[, 4])
sd(clt_sample_k[, 5])
sd(clt_sample_k[, 6])

# or use apply() function to loop
apply(clt_sample_k[, 1:6], 2, sd)
```

3.52158382430107

3.7529497170822

3.57115269611191

4.01026314926038

3.17895630009202

5.01140541270873

3.52158382430107 3.7529497170822 3.57115269611191 4.01026314926038
3.17895630009202 5.01140541270873

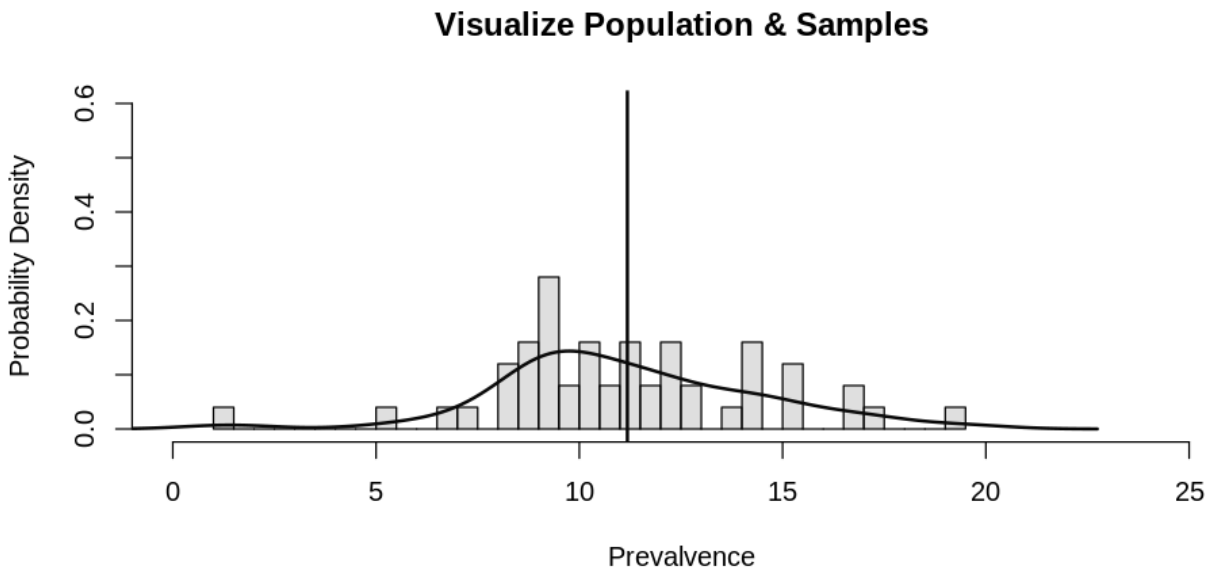
k sample's distributions (k many)

In [291]:
```r
# ---------------------------------
# k sample's distributions (k many)
# ---------------------------------
# Show the first few sample's histogram
par(mfrow=c(2, 3))
apply(clt_sample_k[, 1:6], 2, FUN=hist)
# Reset
par(mfrow=c(1, 1))
```

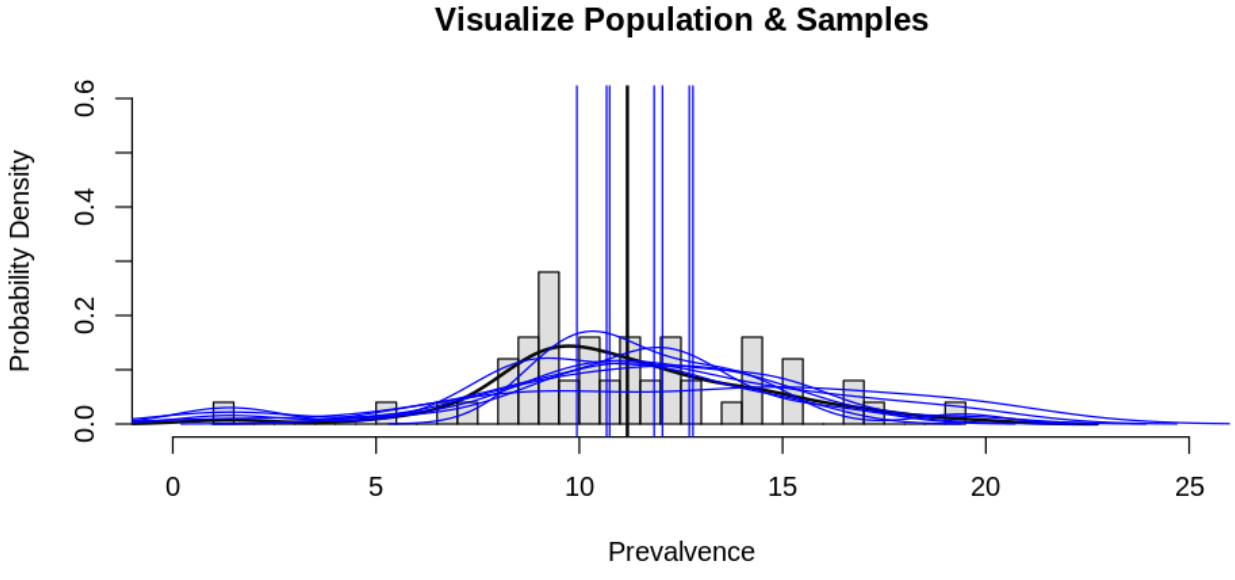
```
[[1]]
$breaks
 [1]  0  2  4  6  8 10 12 14 16 18

$counts
[1] 1 0 1 1 4 5 5 2 1

$density
[1] 0.025 0.000 0.025 0.025 0.100 0.125 0.125 0.050 0.025

$mids
[1]  1  3  5  7  9 11 13 15 17

$xname
[1] "newX[, i]"

$equidist
[1] TRUE

attr( "class")
```

Show the first few sample's density, together with Population

In [292]:
```r
# Show the first few sample's density, together with Population
# Population (Prevalence) histogram in probablity
hist(ASD_State_SPED_2016$Prevalence, probability = T,
     col=rgb(0.75,0.75,0.75,0.5), breaks = 50,
     xlab = 'Prevalvence', xlim = (c(0, 25)),
     ylab = 'Probability Density',
     ylim = (c(0, 0.6)),
     main = 'Visualize Population & Samples')

# Overlay curve:
# Population (Prevalence) density
lines(density(ASD_State_SPED_2016$Prevalence), col="grey4", lwd=2)

# Overlay line:
# mean = mean of Population (Prevalence)
abline(v=mean(ASD_State_SPED_2016$Prevalence), col="grey4", lwd=2)
```



In [293]:
```r
# Show the first few sample's density, together with Population
# Population (Prevalence) histogram in probablity
hist(ASD_State_SPED_2016$Prevalence, probability = T,
     col=rgb(0.75,0.75,0.75,0.5), breaks = 50,
     xlab = 'Prevalvence', xlim = (c(0, 25)),
     ylab = 'Probability Density',
     ylim = (c(0, 0.6)),
     main = 'Visualize Population & Samples')

# Overlay curve:
# Population (Prevalence) density
lines(density(ASD_State_SPED_2016$Prevalence), col="grey4", lwd=2)

# Overlay line:
# mean = mean of Population (Prevalence)
abline(v=mean(ASD_State_SPED_2016$Prevalence), col="grey4", lwd=2)

# Overlay:
# First few sample's density & mean
lines(density(clt_sample_k[, 1]), col="blue", lwd=1)
abline(v=mean(clt_sample_k[, 1]), col="blue", lwd=1)

lines(density(clt_sample_k[, 2]), col="blue", lwd=1)
abline(v=mean(clt_sample_k[, 2]), col="blue", lwd=1)

lines(density(clt_sample_k[, 3]), col="blue", lwd=1)
abline(v=mean(clt_sample_k[, 3]), col="blue", lwd=1)

lines(density(clt_sample_k[, 4]), col="blue", lwd=1)
abline(v=mean(clt_sample_k[, 4]), col="blue", lwd=1)

lines(density(clt_sample_k[, 5]), col="blue", lwd=1)
abline(v=mean(clt_sample_k[, 5]), col="blue", lwd=1)

lines(density(clt_sample_k[, 6]), col="blue", lwd=1)
abline(v=mean(clt_sample_k[, 6]), col="blue", lwd=1)

lines(density(clt_sample_k[, clt_k]), col="blue", lwd=1)
abline(v=mean(clt_sample_k[, clt_k]), col="blue", lwd=1)

# We can see that sample's distributions are all different.
```



[Tips] **We notice that sample's distributions are all different.**

Create Sampling Distribution (only one):

In [294]:
```
# ---------------------------------
# Sampling distribution (only one)
# ---------------------------------
# Calculate sample mean value for k samples
clt_sample_k_mean <- apply(clt_sample_k, 2, mean)
# Show first few sample means
clt_sample_k_mean[1:6]
```

10.745 9.94 12.705 10.67 12.045 12.795

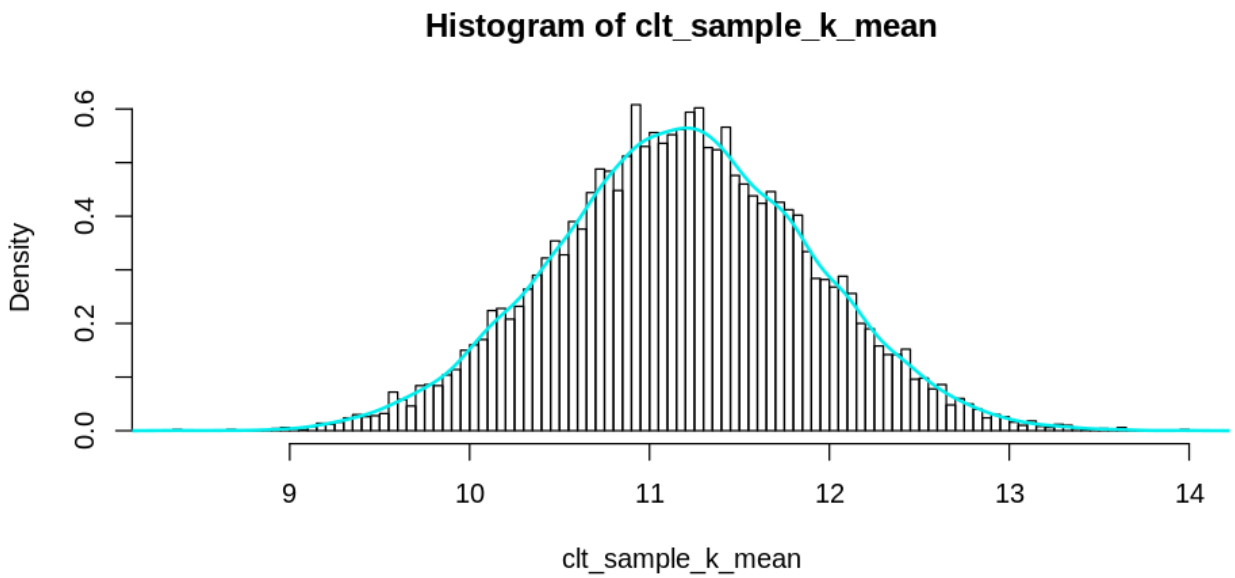
In [295]:
```
# Calculate sample std-dev value for each individual sample (totally k std-dev
clt_sample_k_sd <- apply(clt_sample_k, 2, sd)
# Show first few samples' std-dev
clt_sample_k_sd[1:6]
```

3.52158382430107 3.7529497170822 3.57115269611191 4.01026314926038
3.17895630009202 5.01140541270873

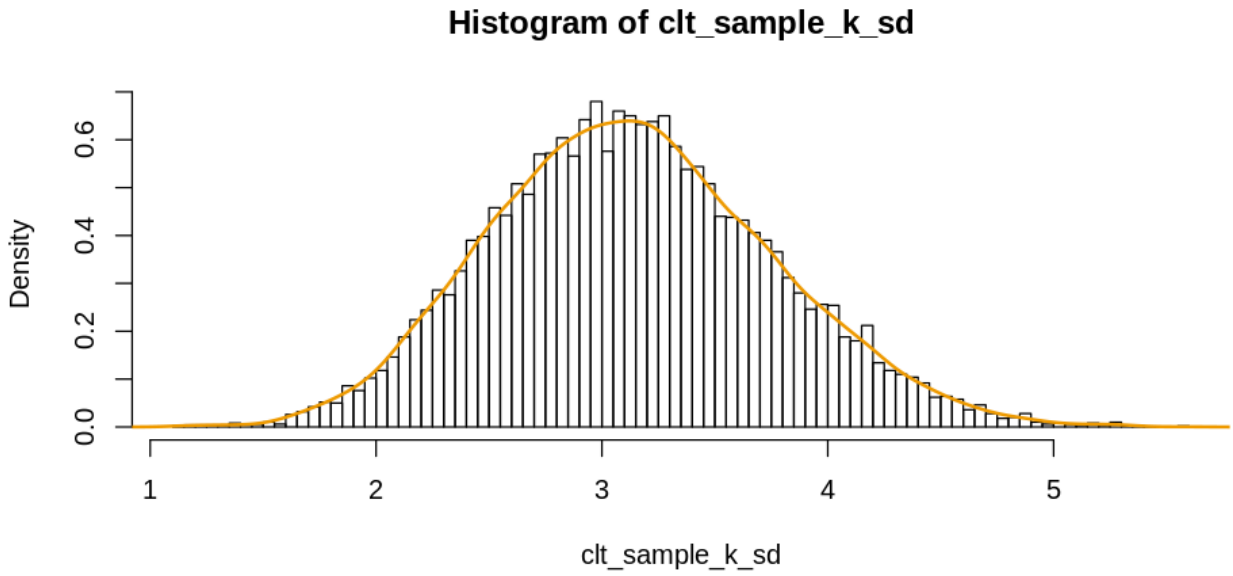
In [296]:
```
# Calculate std-dev value for Sampling DIsdtribution (only one std-dev)
sd(clt_sample_k_mean)
```

0.713336703117785

In [297]:
```
# histogram of sample means (Sampling distribution of the mean)
hist(clt_sample_k_mean, probability = T, breaks = 100)
lines(density(clt_sample_k_mean), col="cyan2", lwd=2)
```



In [298]:
```
# histogram of sample std-dev
hist(clt_sample_k_sd, probability = T, breaks = 100)
lines(density(clt_sample_k_sd), col="orange2", lwd=2)
```



In [299]:
```
# k *Sample* (sample size = n) mean Prevalence
mean(clt_sample_k_mean)
```

11.1788145

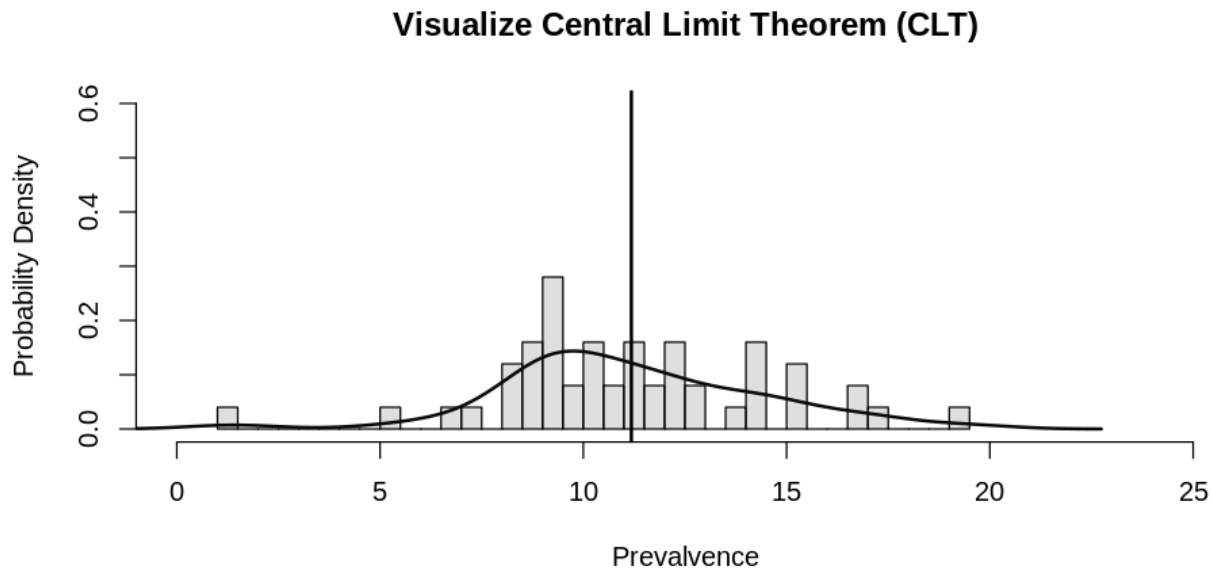
In [300]:
```
# *Population* mean Prevalence
mean(ASD_State_SPED_2016$Prevalence)
```

11.182

[Tips] **We notice that the above two means are close.**

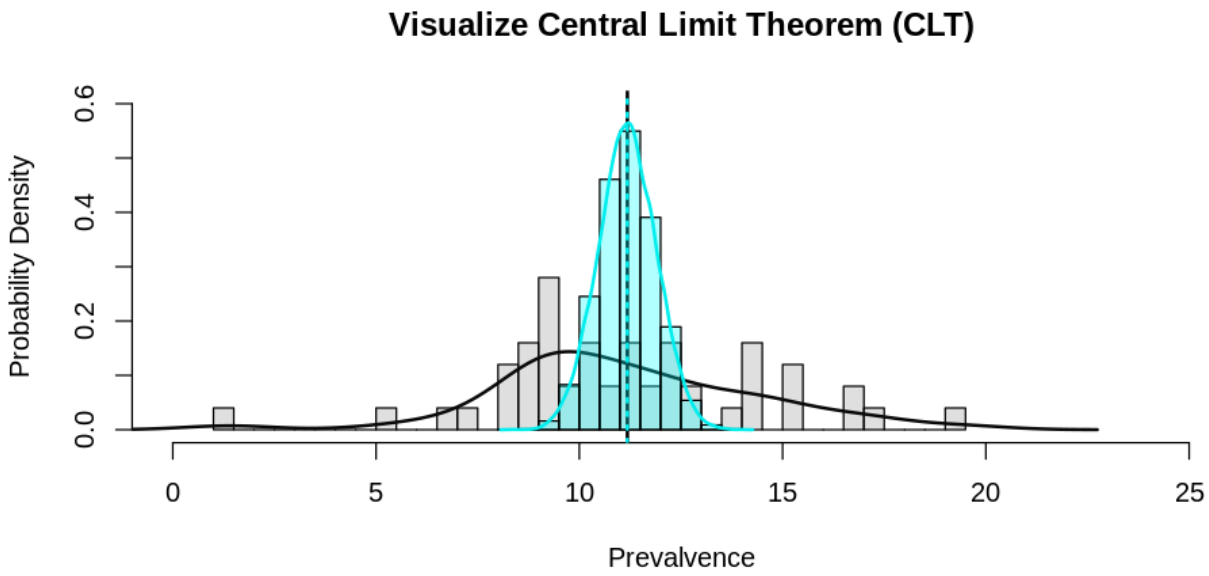
Visualise: Central Limit Theorem (CLT)

In [301]:
```
# ---------------------------------
# Sampling distribution vs. Population distribution vs. Z-Norm
# ---------------------------------
# Create:
# Population (Prevalence) histogram in probablity
hist(ASD_State_SPED_2016$Prevalence, probability = T,
     col=rgb(0.75,0.75,0.75,0.5), breaks = 50,
     xlab = 'Prevalvence', xlim = (c(0, 25)),
     ylab = 'Probability Density', ylim = (c(0, 0.6)),
     main = 'Visualize Central Limit Theorem (CLT)')

# Overlay curve:
# Population (Prevalence) density
lines(density(ASD_State_SPED_2016$Prevalence), col="grey4", lwd=2)

# Overlay line:
# mean = mean of Population (Prevalence)
abline(v=mean(ASD_State_SPED_2016$Prevalence), col="black", lwd=2)
```



Overlay: Sample means histogram in probability (Sampling disribution)

In [302]:
```
# ---------------------------------
# Sampling distribution vs. Population distribution vs. Z-Norm
# ---------------------------------
# Create:
# Population (Prevalence) histogram in probablity
hist(ASD_State_SPED_2016$Prevalence, probability = T,
     col=rgb(0.75,0.75,0.75,0.5), breaks = 50,
     xlab = 'Prevalvence', xlim = (c(0, 25)),
     ylab = 'Probability Density', ylim = (c(0, 0.6)),
     main = 'Visualize Central Limit Theorem (CLT)')

# Overlay curve:
# Population (Prevalence) density
lines(density(ASD_State_SPED_2016$Prevalence), col="grey4", lwd=2)

# Overlay line:
# mean = mean of Population (Prevalence)
abline(v=mean(ASD_State_SPED_2016$Prevalence), col="black", lwd=2)

# Overlay line:
# Sample means histogram in probability (Sampling disribution)
hist(clt_sample_k_mean, probability = T,
     col=rgb(0,1,1,0.3), # https://www.dataanalytics.org.uk/make-transparent-c
     add=T)

# Overlay curve:
# Sample (Prevalence) density (Sampling disribution)
lines(density(clt_sample_k_mean), col="cyan2", lwd=2)
# Overlay line:
# mean of Sampling distribution (of Prevelance, sample size n)
abline(v=mean(clt_sample_k_mean), col="cyan2", lwd=2, lty=3)
```



< How to make transparent colors in R > https://www.dataanalytics.org.uk/make-transparent-colors-in-r/ (https://www.dataanalytics.org.uk/make-transparent-colors-in-r/)

In [303]:
```
col2rgb(c("cyan", "grey", "purple", "orange")) / 255
```

red	0	0.745098	0.6274510	1.0000000
green	1	0.745098	0.1254902	0.6470588
blue	1	0.745098	0.9411765	0.0000000

In [304]:
```
# Recall:
# k *Sample* (sample size = n) mean Prevalence
mean(clt_sample_k_mean)
# *Population* mean Prevalence
mean(ASD_State_SPED_2016$Prevalence)
# We see that the above two means are close. Good estimation!
```

11.1788145

11.182

[Tips] We notice that the above two means are close. Good estimation!

Standard Error (SE) (of mean prevalence), can be estimated as: std-dev of the Sampling distribution (of mean prevalence):

In [305]:
```
# ---------------------------------
# Standard Error (SE) (of mean prevalence), can be estimated as:
# std-dev of the Sampling distribution (of mean prevalence)
# ---------------------------------
# https://en.wikipedia.org/wiki/Sampling_distribution

# [1] Actual SE: When Population std-dev is known, SE using Population standar
sd.p(ASD_State_SPED_2016$Prevalence) / sqrt(clt_n)

# [2] Estimated SE with k samples: When Population std-dev is NOT known, but m
sd(clt_sample_k_mean)

# [3] Estimated SE with only one sample: When Population std-dev is NOT known,
clt_sample_k_sd[1] / sqrt(clt_n)
```

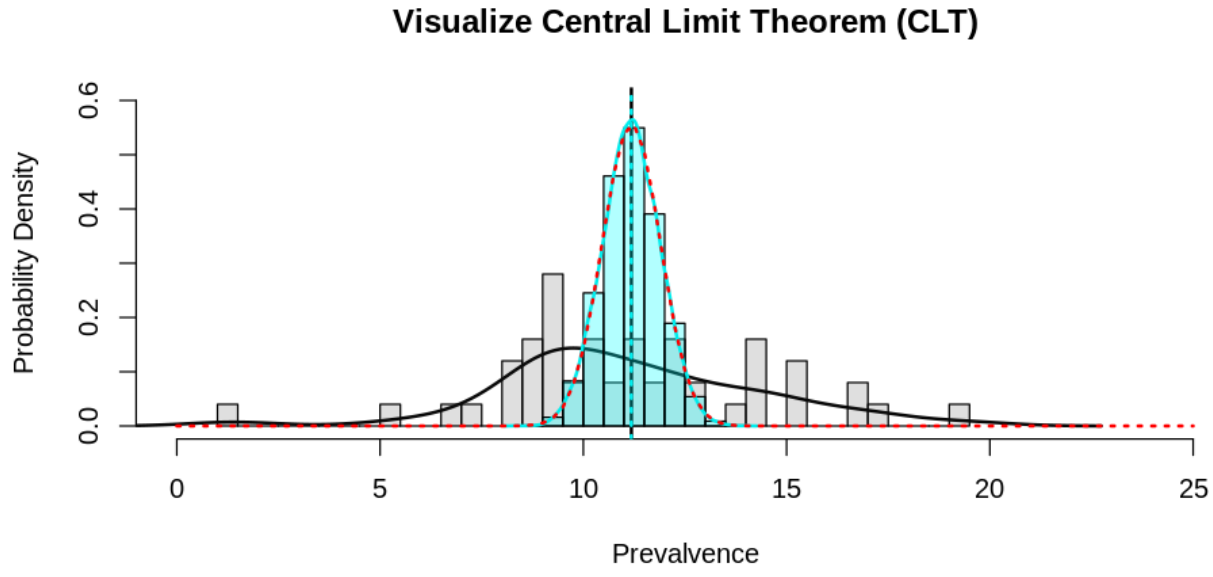
0.715705106870141

0.713336703117785

0.787450081960087

Overlay: Theoretic Sampling Distribution with population mean & std-dev = Actual SE:

In [306]:
```r
# ---------------------------------
# Sampling distribution vs. Population distribution vs. Z-Norm
# ---------------------------------
# Create:
# Population (Prevalence) histogram in probablity
hist(ASD_State_SPED_2016$Prevalence, probability = T,
     col=rgb(0.75,0.75,0.75,0.5), breaks = 50,
     xlab = 'Prevalvence', xlim = (c(0, 25)),
     ylab = 'Probability Density', ylim = (c(0, 0.6)),
     main = 'Visualize Central Limit Theorem (CLT)')

# Overlay curve:
# Population (Prevalence) density
lines(density(ASD_State_SPED_2016$Prevalence), col="grey4", lwd=2)

# Overlay line:
# mean = mean of Population (Prevalence)
abline(v=mean(ASD_State_SPED_2016$Prevalence), col="black", lwd=2)

# Overlay line:
# Sample means histogram in probability (Sampling disribution)
hist(clt_sample_k_mean, probability = T,
     col=rgb(0,1,1,0.3), # https://www.dataanalytics.org.uk/make-transparent-c
     add=T)

# Overlay curve:
# Sample (Prevalence) density (Sampling disribution)
lines(density(clt_sample_k_mean), col="cyan2", lwd=2)
# Overlay line:
# mean of Sampling distribution (of Prevelance, sample size n)
abline(v=mean(clt_sample_k_mean), col="cyan2", lwd=2, lty=3)

# Overlay curve:
# *Theoretic Sampling Distribution* with population mean & std-dev = Actual SE
# mean = mean of Population (Prevalence) & std-dev = std-dev of Population (Pr
curve(dnorm(x,
            mean(ASD_State_SPED_2016$Prevalence), # Actual Population mean
            sd.p(ASD_State_SPED_2016$Prevalence) / sqrt(clt_n)), # Actual SE (
      add=TRUE, col="red", lwd=2, lty=3)
```

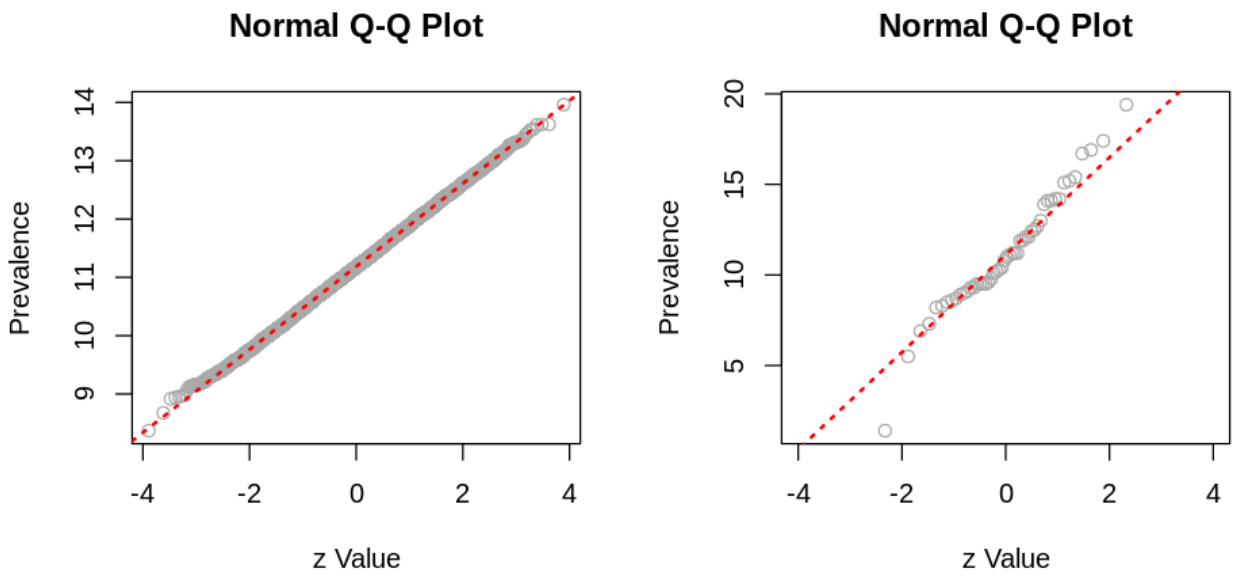


Sampling & Normality - Evaluate Normality

Use QQ Plot

< Construct a Quantile-Quantile Plot (QQ plot) > https://youtu.be/okjYjClSjOg (https://youtu.be/okjYjClSjOg)

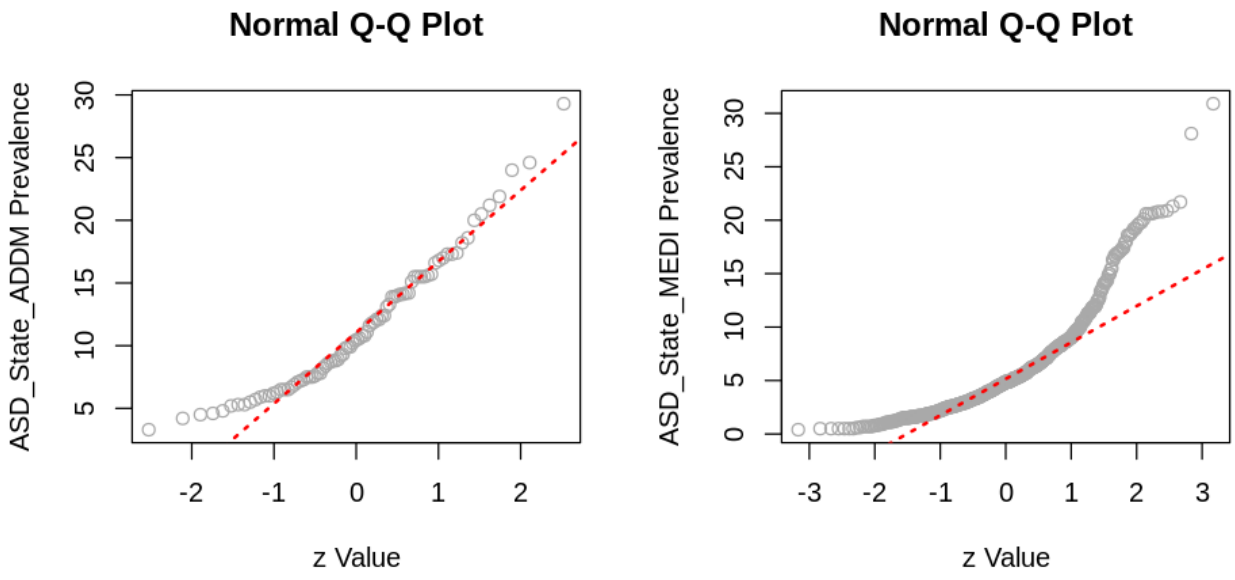
In [307]:
```
# ---------------------------------
# Evaluate normality
# ---------------------------------
# Construct a Quantile-Quantile Plot (QQ plot)
# https://youtu.be/okjYjClSjOg

par(mfrow=c(1, 2))
# Sample means
qqnorm(clt_sample_k_mean, col="darkgrey",
#       xlim=(c(-4, 4)), ylim=(c(0, 20)),
       xlab="z Value", ylab="Prevalence")
qqline(clt_sample_k_mean, col="red", lwd=2, lty=3)
# Population
qqnorm(ASD_State_SPED_2016$Prevalence, col="darkgrey",
       xlim=(c(-4, 4)),
       xlab="z Value", ylab="Prevalence")
qqline(ASD_State_SPED_2016$Prevalence, col="red", lwd=2, lty=3)
# Reset
par(mfrow=c(1, 1))
```

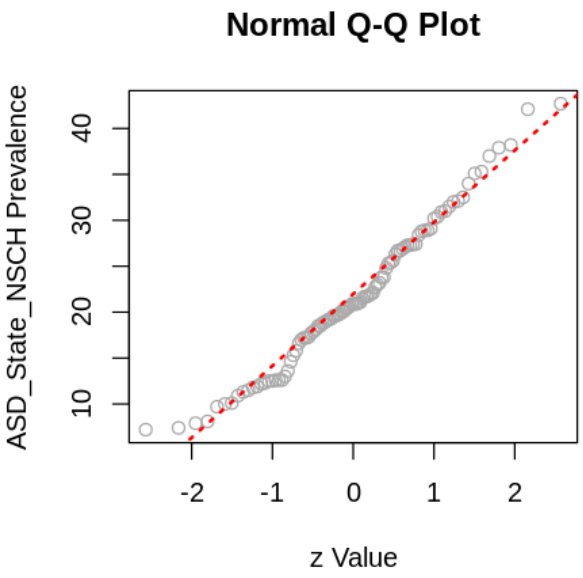


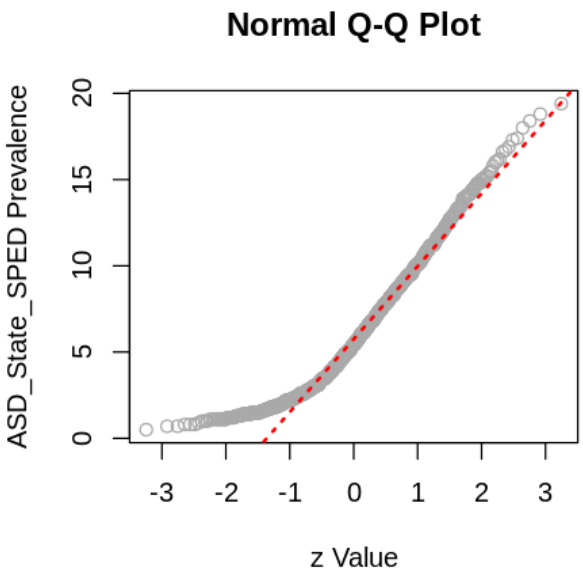
[Tips] If most/all data points are aligned with the red straight line, then the underlying data points are normally distributed.

In [308]:

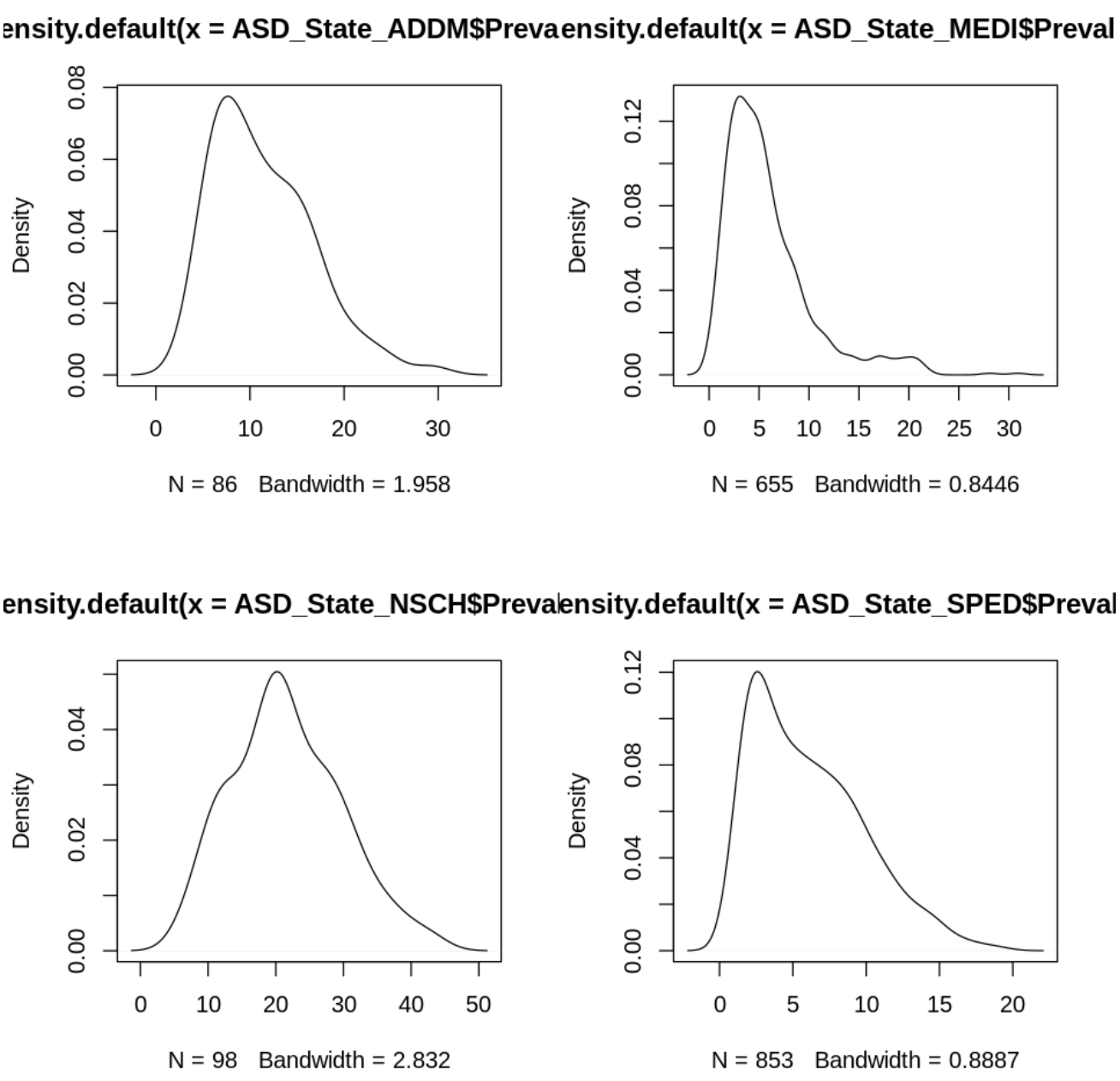
```
# ---------------------------------
# Evaluate normality
# ---------------------------------
par(mfrow=c(1, 2))

# ASD_State_ADDM$Prevalence
qqnorm(ASD_State_ADDM$Prevalence, col="darkgrey",
#       xlim=(c(-4, 4)), ylim=(c(0, 20)),
       xlab="z Value", ylab="ASD_State_ADDM Prevalence")
qqline(ASD_State_ADDM$Prevalence, col="red", lwd=2, lty=3)
# plot(density(ASD_State_ADDM$Prevalence))

# ASD_State_MEDI$Prevalence
qqnorm(ASD_State_MEDI$Prevalence, col="darkgrey",
#       xlim=(c(-4, 4)), ylim=(c(0, 20)),
       xlab="z Value", ylab="ASD_State_MEDI Prevalence")
qqline(ASD_State_MEDI$Prevalence, col="red", lwd=2, lty=3)

# ASD_State_NSCH$Prevalence
qqnorm(ASD_State_NSCH$Prevalence, col="darkgrey",
#       xlim=(c(-4, 4)), ylim=(c(0, 20)),
       xlab="z Value", ylab="ASD_State_NSCH Prevalence")
qqline(ASD_State_NSCH$Prevalence, col="red", lwd=2, lty=3)

# ASD_State_SPED$Prevalence
qqnorm(ASD_State_SPED$Prevalence, col="darkgrey",
#       xlim=(c(-4, 4)), ylim=(c(0, 20)),
       xlab="z Value", ylab="ASD_State_SPED Prevalence")
qqline(ASD_State_SPED$Prevalence, col="red", lwd=2, lty=3)

# Reset
par(mfrow=c(1, 1))
```




Normal Q-Q Plot
ASD_State_NSCH Prevalence
10
20
30
40
-2
-1
0
1
2
z Value


Normal Q-Q Plot
ASD_State_SPED Prevalence
0
5
10
15
20
-3
-2
-1
0
1
2
3
z Value

In [309]:
```
# ---------------------------------
# Evaluate normality
# ---------------------------------
par(mfrow=c(1, 2))
plot(density(ASD_State_ADDM$Prevalence))
plot(density(ASD_State_MEDI$Prevalence))
plot(density(ASD_State_NSCH$Prevalence))
plot(density(ASD_State_SPED$Prevalence))
# Reset
par(mfrow=c(1, 1))
```



In [310]:
```
# Alternatively, use shapiro.test() to test Normality
set.seed(88)

# Test data of k smaple's means (Sampling Distribution data):
shapiro.test(sample(x = clt_sample_k_mean, size = 1000))

# Test data of population's Prevalence values (Population Distribution data):
shapiro.test(ASD_State_SPED_2016$Prevalence)
```

```
        Shapiro-Wilk normality test

data:  sample(x = clt_sample_k_mean, size = 1000)
W = 0.99904, p-value = 0.8907


        Shapiro-Wilk normality test

data:  ASD_State_SPED_2016$Prevalence
W = 0.96985, p-value = 0.2282
```

[Tips] General speaking, if p-value is greater than 0.05 (meaning more than 5% chance of being normally distributed), then the underlying data points are normally distributed.

Confidence Interval (CI)

Confidence Interval (CI) - Mean Estimation & Its CI

Use a Sample statistic (e.g. mean) to estimate a population statistic (e.g. mean). And quantitatively calculate the confidence of the estimation.

In [311]:
```
# ---------------------------------
# Use a sample of a few US. State's ASD prevalence (mean) to estimate:
# Average prevalence of ALL US. States (the *Population*) [Source SPED, Year 2
# ---------------------------------
dim(ASD_State_SPED_2016)
#
ASD_State_SPED_2016 # This is considered as a population now.
```

50 2

State	Prevalence
AL	9.1
AK	10.1
AZ	10.4
AR	9.5
CA	13.9
CO	7.3
CT	15.4
DE	11.1
DC	11.9
FL	12.1
GA	10.3
HI	8.3
ID	9.5
IL	11.0
IN	14.2
IA	1.4
KS	8.2
KY	9.3
LA	6.9
ME	16.7
MD	11.9
MA	17.4
MI	11.2
MN	19.4
MS	9.5
MO	12.4
MT	5.5
NE	10.8
NV	12.7
NH	14.1
NJ	14.1
NM	8.5
NY	13.0
NC	11.2
ND	9.8

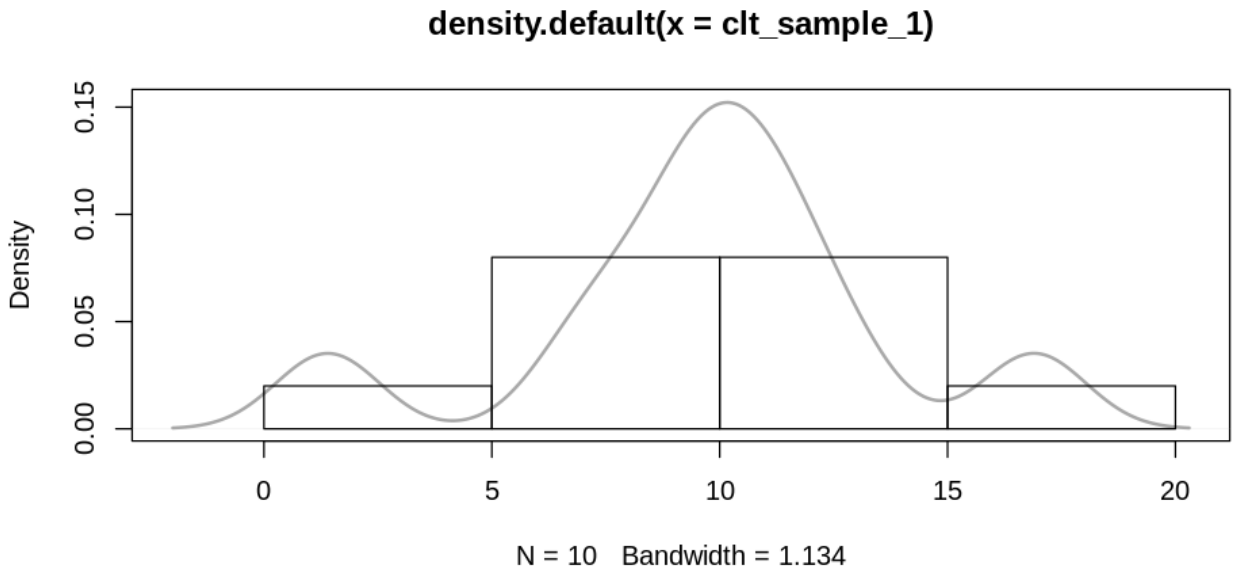
State	Prevalence
OH	12.5
OK	8.9
OR	15.1
PA	16.9
RI	15.2
SC	9.6
SD	9.0
TN	9.5
TX	10.2
UT	8.7
VT	12.1
VA	14.2
WA	11.2
WV	8.6
WY	9.3

Draw a Sample from ASD_State_SPED_2016$Prevalence

In [312]:
```
# Create a *Sample* from ASD_State_SPED_2016$Prevalence,
# with sample size n =
clt_n = 10
# Try 20 or 40, larger sample size, narrower the CI (more confident at xx% lev
# clt_n = 20

set.seed(88)
clt_sample_1 = sample(x = ASD_State_SPED_2016$Prevalence, size = clt_n, replac
clt_sample_1
```

11.2 9.5 16.9 6.9 11.2 8.2 12.7 9.5 1.4 10.2

In [313]:
```
plot(density(clt_sample_1), col="darkgrey", lwd=2)
hist(clt_sample_1, probability = T, add = T)
```



In [314]:
```
# Sample mean Prevalence
mean(clt_sample_1)

# *Population* mean Prevalence
mean(ASD_State_SPED_2016$Prevalence)
```

9.77

11.182

1. Calculate Confidence Interval of mean estimation: CI using Z (Standard Normal) distribution

In [315]:
```
# ---------------------------------
# CI using Z (Standard Normal) distribution
# ---------------------------------
# sample mean
sample_mean = mean(clt_sample_1)
sample_mean
```

9.77

In [316]:
```
# sample size n
sample_size_n = length(clt_sample_1)
sample_size_n
```

10

In [317]:
```
# sample standard deviation
sample_sd = sd(clt_sample_1)
sample_sd
```

4.00833853083516

In [318]:
```
# sample standard error
sample_se = sample_sd / sqrt(sample_size_n)
sample_se
```

1.26754793904522

In [319]:
```
# 95% quantile (z score)
z_score = qnorm(p = 0.975)
z_score
```

1.95996398454005

In [320]:
```
# ?qnorm
```

In [321]:
```
# CI using Z distribution
sample_ci = z_score * sample_se
sample_ci
```

2.4843483092066

In [322]:
```
# Lower CI: mean + CI
sample_mean - sample_ci

# Upper CI: mean + CI
sample_mean + sample_ci

# Display
cat('\t< Confidence Interval (Prevalence) >\n',  '\tLower CI : ',  sample_mean
```

7.2856516907934

12.2543483092066

```
        < Confidence Interval (Prevalence) >
        Lower CI :  7.285652    Mean :  9.77    Upper CI :  12.25435
```

[?] Is the population mean in this CI range?

2. Calculate Confidence Interval of mean estimation: CI using T distribution

In [323]:
```
# ---------------------------------
# CI using T distribution
# ---------------------------------
# sample mean
sample_mean = mean(clt_sample_1)
sample_mean
# sample size n
sample_size_n = length(clt_sample_1)
sample_size_n
# sample standard deviation
sample_sd = sd(clt_sample_1)
sample_sd
# sample standard error
sample_se = sample_sd / sqrt(sample_size_n)
sample_se
```

9.77

10

4.00833853083516

1.26754793904522

In [324]:
```
# 95% quantile (t score)
t_score = qt(p = 0.975, df = sample_size_n - 1)
t_score
```

2.2621571627982

In [325]:
```
# ?qt
```

In [326]:
```
# CI using T distribution
sample_ci = t_score * sample_se
sample_ci

# Lower CI: mean + CI
sample_mean - sample_ci

# Upper CI: mean + CI
sample_mean + sample_ci

# Display
cat('\t< Confidence Interval (Prevalence) >\n',  '\tLower CI : ',  sample_mean
```

2.86739264950124

6.90260735049876

12.6373926495012

```
        < Confidence Interval (Prevalence) >
        Lower CI :  6.902607    Mean :  9.77    Upper CI :  12.63739
```

[?] Is the population mean in this CI range?

[?] Compare CIs of Z and T distribution, which CI has wider range? Is it reasonable?

In [327]:
```
# Alternatively, calculate CI using t.test() function
t.test(clt_sample_1, conf.level = 0.95)
```

```
        One Sample t-test

data:  clt_sample_1
t = 7.7078, df = 9, p-value = 2.976e-05
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
  6.902607 12.637393
sample estimates:
mean of x
     9.77
```

In [328]:
```
# Two group hypothesis test : sample mean vs. population mean
t.test(clt_sample_1, conf.level = 0.95, mu = mean(ASD_State_SPED_2016$Prevalen
```

```
        One Sample t-test

data:  clt_sample_1
t = -1.114, df = 9, p-value = 0.2942
alternative hypothesis: true mean is not equal to 11.182
95 percent confidence interval:
  6.902607 12.637393
sample estimates:
mean of x
     9.77
```

Quiz:

Obtain CI using smaller/larger sample size (clt_n) at 99% confidence. Compare CI width.

Observe: larger sample size, narrower the CI (more confident at xx% level)

In [329]:
```
# Write your code below and press Shift+Enter to execute
```

Double-click **here** for the solution.

Confidence Interval (CI) - Proportion Estimation & Its CI

Use a Sample statistic (e.g. proportioin) to estimate a population statistic (e.g. proportioin). And quantitatively calculate the confidence of the estimation.

In [330]:
```
# ---------------------------------
# Use a sample of one US. State's ASD prevalence (proportion) to estimate:
# Prevalence of THAT US. State's ALL Children (the *Population*) [Source SPED,
# ---------------------------------

# No. Children with ASD
ASD  <- ASD_State_SPED$Numerator_ASD[ASD_State_SPED$Year == 2016]
#
str(ASD)
```

```
 int [1:50] 6140 1204 10746 4181 79041 5902 7391 1383 782 30920 ...
```

In [331]:
```
# No. Children with ASD of first US. State (AL-Alabama)
ASD[1]
```

```
6140
```

In [332]:
```
# No. Children surveyed
Children  <- ASD_State_SPED$Denominator[ASD_State_SPED$Year == 2016]
#
str(Children)
```

```
 int [1:50] 674701 119217 1033241 440130 5686400 808556 479961 124609 65732 2
555399 ...
```

In [333]:
```
# No. Children surveyed of first US. State (AL-Alabama)
Children[1]
```

```
674701
```

1. Calculate Confidence Interval of proportion estimation: CI using Z score interval (standard normal distribution)

In [334]:
```
# ---------------------------------
# CI using Z score interval  (standard normal distribution)
# https://en.wikipedia.org/wiki/Binomial_proportion_confidence_interval
# ---------------------------------

# sample proportion of first US. State (AL-Alabama) in year 2016 of source SPE
sample_proportion = ASD[1] / Children[1]
sample_proportion # p
1- sample_proportion # q = 1 - p
```

0.00910032740428723

0.990899672595713

In [335]:
```
# sample size n
sample_size_n = Children[1]
sample_size_n
```

674701

In [336]:
```
# 95% quantile (z score)
z_score = qnorm(p = 0.975)
z_score
```

1.95996398454005

In [337]:
```
sample_ci = z_score * sqrt(sample_proportion * (1 - sample_proportion) / sampl
sample_ci
```

0.000226587404757579

In [338]:
```
# Lower CI: mean + CI
sample_proportion - sample_ci

# Upper CI: mean + CI
sample_proportion + sample_ci

# Display
cat('\t< Confidence Interval >\n',  '\tLower CI : ',  sample_proportion - samp
```

0.00887373999952965

0.00932691480904481

```
        < Confidence Interval >
        Lower CI :  0.00887374  Mean :  0.009100327     Upper CI :  0.0093269
15
```

In [339]:
```
# Display * 1000 -> Prevalence
cat('\t< Confidence Interval (Prevalence) >\n',  '\tLower CI : ',  1000*(sampl
```

```
        < Confidence Interval (Prevalence) >
        Lower CI :  8.87374     Mean :  9.100327        Upper CI :  9.326915
```

[Tips] Based above calculation upon [Source: SPED] [Year: 2016] data, we have 95% confidence that: The actual AL-Alabama state level ASD prevalence (if ALL childrens in Alabama state were surveyed) would be in the above calculated CI range 95% times.

Or, asumming there are 100 different Alabama states exist in 100 parallel universes, we obtained 100 actual prevelance proportions. 95 of them will likely fall into the CI range.

2. Calculate Confidence Interval of proportion estimation: CI using Wilson score interval

In [340]:
```
# ---------------------------------
# CI using Wilson score interval
# https://en.wikipedia.org/wiki/Binomial_proportion_confidence_interval
# ---------------------------------

sample_proportion # = ASD[1] / Children[1]

# Yates' chi-squared test = Wilson score interval with continuity correction -
prop.test(ASD[1], Children[1], conf.level = 0.95)

# Pearson's chi-squared test = Wilson score interval - wilson
prop.test(ASD[1], Children[1], conf.level = 0.95, correct = FALSE)
```

0.00910032740428723

```
        1-sample proportions test with continuity correction

data:  ASD[1] out of Children[1], null probability 0.5
X-squared = 650363, df = 1, p-value < 2.2e-16
alternative hypothesis: true p is not equal to 0.5
95 percent confidence interval:
 0.008875786 0.009330477
sample estimates:
          p
0.009100327


        1-sample proportions test without continuity correction

data:  ASD[1] out of Children[1], null probability 0.5
X-squared = 650365, df = 1, p-value < 2.2e-16
alternative hypothesis: true p is not equal to 0.5
95 percent confidence interval:
 0.008876518 0.009329726
sample estimates:
          p
0.009100327
```

Quiz:

Obtain CI of Male.Prevalence propotion [Source: ADDM] [Year: 2014] at 99% confidence.

In [341]:
```
# Write your code below and press Shift+Enter to execute
```

Double-click **here** for the solution.

Quiz:

Obtain CI of Female.Prevalence propotion [Source: ADDM] [Year: 2014] at 99% confidence.

Then Compare CI range with Male children's CI range. Which gender has statistically higer ASD prevalence/proportion?

In [342]:
```
# Write your code below and press Shift+Enter to execute
```

Double-click **here** for the solution.

```
()
```

Workshop Submission

What to submit?

Choose one of below visualisations/charts, use R to construct the chart nicely.

Optionally, enhance it with additional data dimensions to be better than original chart.

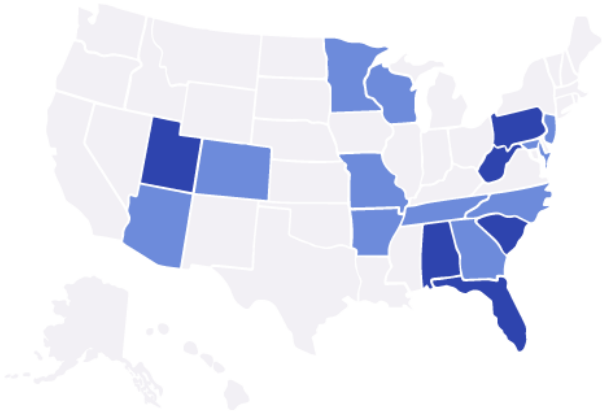
https://www.cdc.gov/ncbddd/autism/data/index.html (https://www.cdc.gov/ncbddd/autism/data/index.html)

Select data source: ADDM Network*

ASD Data Collection Locations for: ADDM Network*

Since the launch of the ADDM Network in 2000, CDC has funded **16 sites** at various times. In 2014, ASD data were collected from **11 sites** by obtaining the health and education records of children with behaviors consistent with ASD.

ASD Collection Sites



No Participation or Data Suppressed

Available Data from Most Recent Data Collection

Available Data from Previous Data Collection

WHY THIS MATTERS

When reviewing ASD data and findings, it is important to consider *where* the data were collected and *how* each location might affect the data. Across the United States, each community has a unique population with different characteristics. There are also regional differences in healthcare and education systems, which can affect *when* and *how* children with ASD are identified, as well as the services they receive.

Because of these geographic differences, it may not be possible to directly compare data collected in one community to data from other communities. Take ADDM Network data collected from 11 sites in 2014, for example. In Colorado, ASD prevalence was 13.9 out of 1,000 kids, whereas in North Carolina, ASD prevalence was 17.4 out of 1,000 kids. There is a clear difference in the number of children identified with ASD in these two states, but without additional information, it is difficult to know *why* these differences exist. Therefore, it would not be correct to assume that the prevalence in one state will be the same as another state.

**ADDM data do not represent the entire state, only a selection of sites within the state.*

2014 ADDM NETWORK DATA

In this section, explore the most recent ADDM data, both overall and among certain demographic groups by study area.

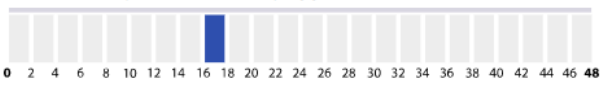
Select a location: U.S. or Total†

MOST RECENT STUDY YEAR: 2014

ASD PREVALENCE PER 1,000 8-YEAR-OLD CHILDREN

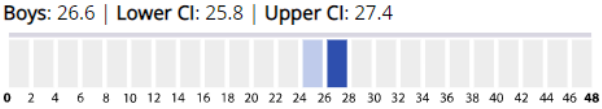
Prevalence **Overall**

Overall: 16.8 | **Lower CI**: 16.4 | **Upper CI**: 17.3

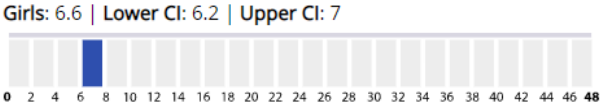


Prevalence **By Sex**

Boys: 26.6 | **Lower CI**: 25.8 | **Upper CI**: 27.4

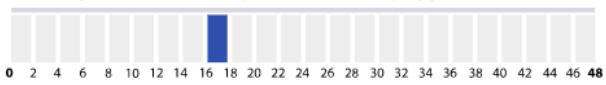


Girls: 6.6 | **Lower CI**: 6.2 | **Upper CI**: 7

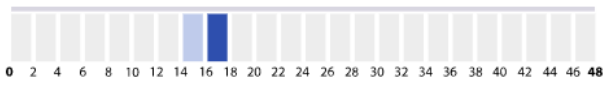


Prevalence **By Race/Ethnicity**

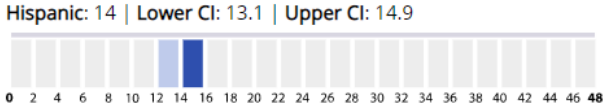
Non-Hispanic White: 17.2 | **Lower CI**: 16.5 | **Upper CI**: 17.8



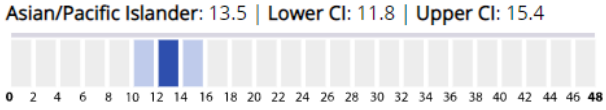
Non-Hispanic Black: 16 | **Lower CI**: 15.1 | **Upper CI**: 16.9



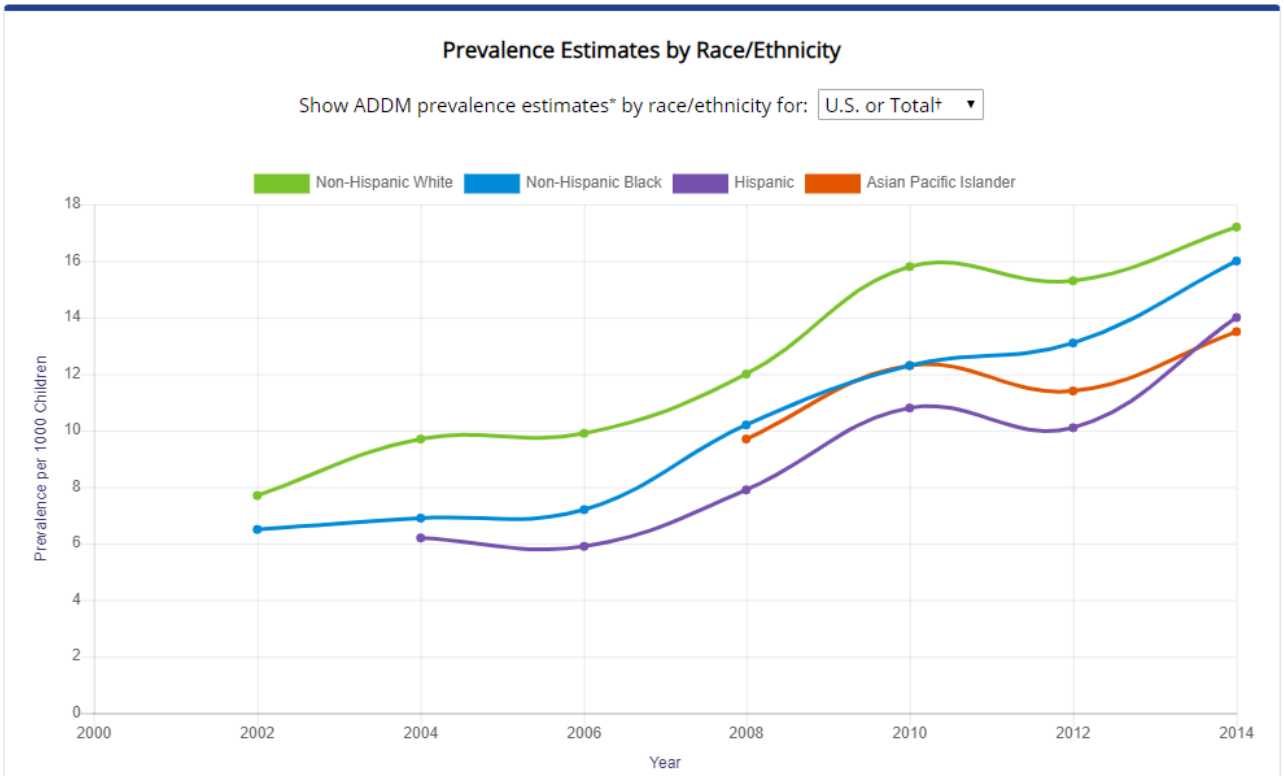
Hispanic: 14 | **Lower CI**: 13.1 | **Upper CI**: 14.9



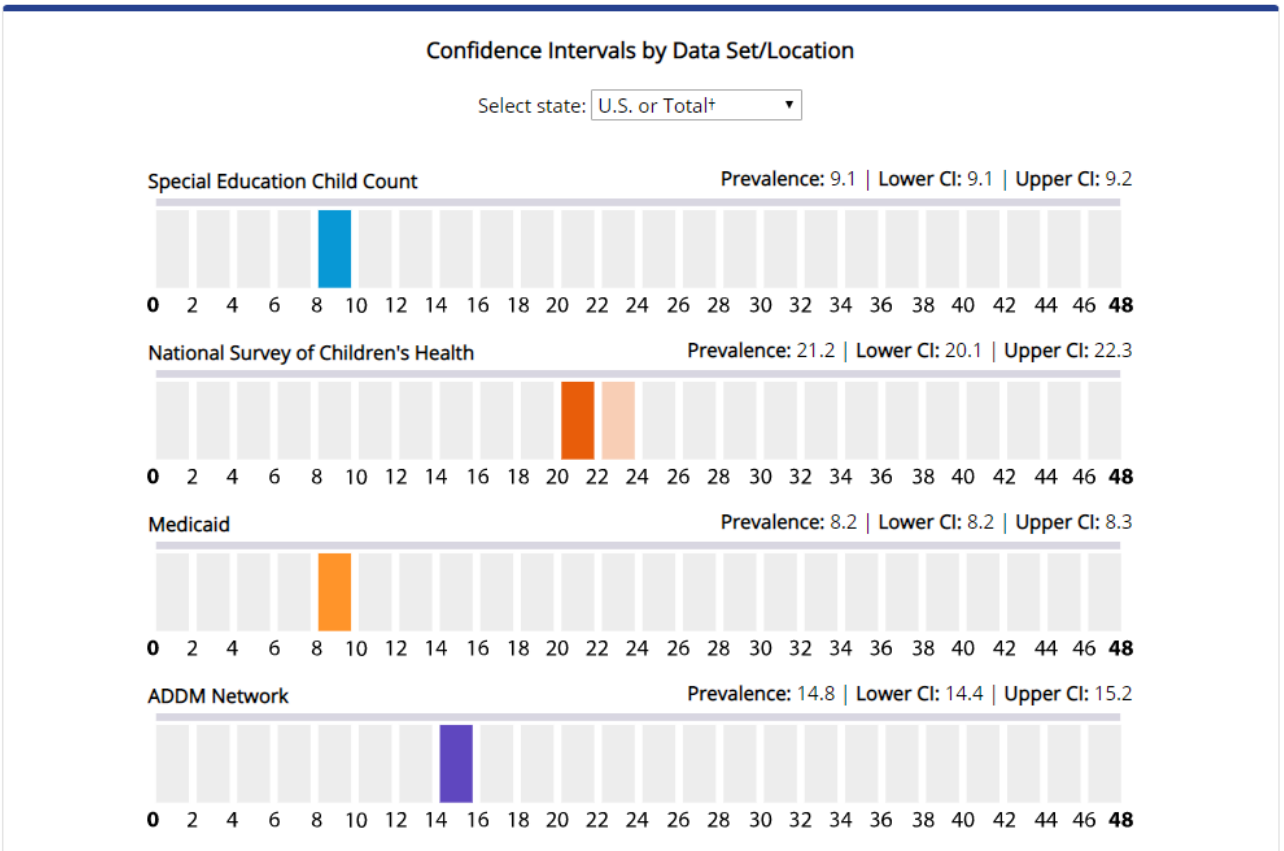
Asian/Pacific Islander: 13.5 | **Lower CI**: 11.8 | **Upper CI**: 15.4



†ADDM estimate = the total for all sites combined.



Note: *Click the icons and racial/ethnic groups above the chart to hide or unhide data. Hover your mouse over data points to show prevalence by year.*
**ADDM data do not represent the entire state, only a selection of sites within the state.*
†ADDM estimate = the total for all sites combined.



WHY THIS MATTERS

By comparing different data sets, we see that some confidence intervals are wide, while others are narrow. When a confidence interval is wide, the true prevalence may be anywhere within that range, making it less certain. A narrow confidence interval means we can be more certain about the reported prevalence.

Note: *The graph above shows data from 2012, the most recent year for which all data sets had data.*
†ADDM estimate = the total for all sites combined.

In [343]:
```
# Write your code below and press Shift+Enter to execute
```

Excellent! You have completed the workshop notebook!

Connect with the author:

This notebook was written by GU Zhan (Sam) (https://sg.linkedin.com/in/zhan-gu-27a82823).

Sam (https://www.iss.nus.edu.sg/about-us/staff/detail/201/GU_Zhan) is currently a lecturer in Institute of Systems Science (https://www.iss.nus.edu.sg/) in National University of Singapore (http://www.nus.edu.sg/). He devotes himself into pedagogy & andragogy, and is very passionate in inspiring next generation of artificial intelligence lovers and leaders.

()

Appendices

Interactive workshops: < Learning R inside R > using swirl() (in R/RStudio)

https://github.com/telescopeuser/S-SB-Workshop (https://github.com/telescopeuser/S-SB-Workshop)

Neural Network 101 using nnet()

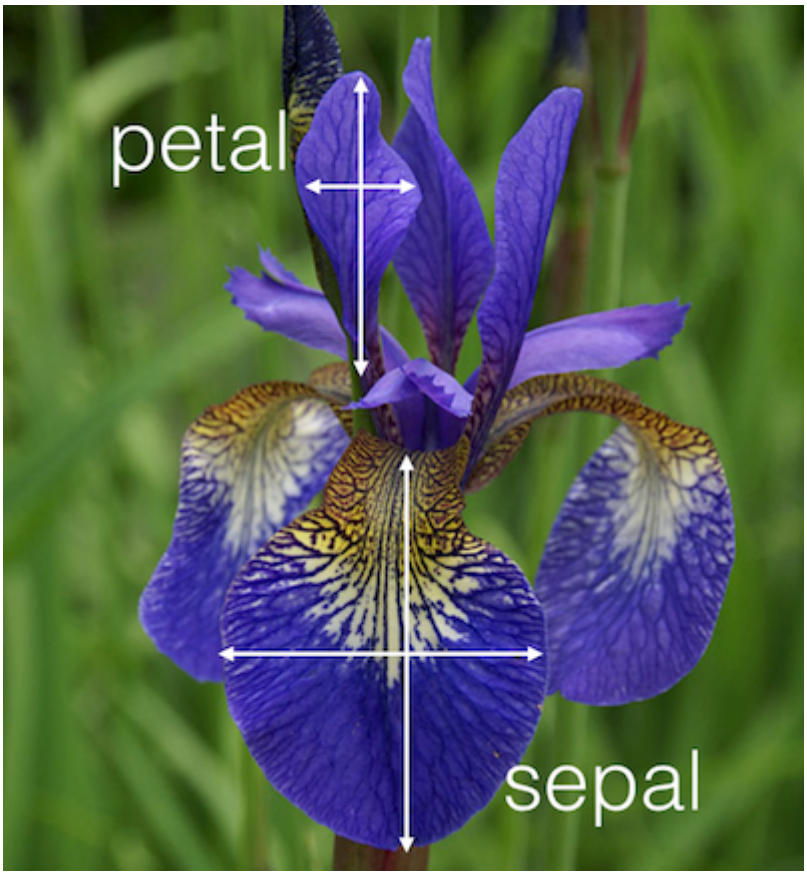
Use nerual net to classify three different species of iris flowers, based on four features/measurements of:

- length of the petals
- width of the petals
- length of the sepals
- width of the sepals

Iris setosa

Iris versicolor

Iris virginica



In [344]:
```r
# ---------------------------------
# Neural Network 101 using nnet()
# ---------------------------------
if(!require(nnet)){install.packages("nnet")}
library("nnet")
# ?nnet

# < Case: predict three different iris flower types >

# https://en.wikipedia.org/wiki/Iris_flower_data_set
# https://archive.ics.uci.edu/ml/datasets/iris

# Data preparation: split iris data in two halves, for training & testing resp
ir <- rbind(iris3[,,1],iris3[,,2],iris3[,,3])
targets <- class.ind( c(rep("setosa", 50), rep("versicolor", 50), rep("virgini
samp <- c(sample(1:50,25), sample(51:100,25), sample(101:150,25))
# Model training (machine learning / data fitting)
ir1 <- nnet(ir[samp,], targets[samp,], size = 2, rang = 0.1,
            decay = 5e-4, maxit = 200)
# Model evaluation function
test.cl <- function(true, pred) {
  true <- max.col(true)
  cres <- max.col(pred)
  table(true, cres)
}
# Model evaluation
test.cl(targets[-samp,], predict(ir1, ir[-samp,]))
```

```
Loading required package: nnet

# weights:  19
initial  value 56.632729
iter  10 value 34.513425
iter  20 value 21.757237
iter  30 value 17.318434
iter  40 value 16.805235
iter  50 value 16.694727
iter  60 value 16.665706
iter  70 value 16.534461
iter  80 value 2.909665
iter  90 value 1.459194
iter 100 value 0.977873
iter 110 value 0.740445
iter 120 value 0.588337
iter 130 value 0.451657
iter 140 value 0.374631
iter 150 value 0.359837
iter 160 value 0.352404
iter 170 value 0.348566
iter 180 value 0.347337
iter 190 value 0.347286
iter 200 value 0.347267
final  value 0.347267
stopped after 200 iterations

    cres
true  1  2  3
   1 25  0  0
   2  0 21  4
   3  0  0 25
```

Correlation of Numeric Variables

In [345]:
```
# ---------------------------------
# Correlation of Numeric Variables
# ---------------------------------
cor_df = select_if(ASD_State, is.numeric) # Select only numeric variables
cor_df = cor_df[, colSums(is.na(cor_df)) == 0] #  Select vaariables without NA

# Compute correlation matrix for No-NA numeric variables:
cor_table = cor(cor_df)
cor_table
```

	Denominator	Prevalence	Lower.CI	Upper.CI	Year	Numerato
Denominator	1.00000000	-0.1374662	-0.07863304	-0.17389486	0.02851671	0.82
Prevalence	-0.13746621	1.0000000	0.95813468	0.96568034	0.64002950	0.11
Lower.CI	-0.07863304	0.9581347	1.00000000	0.85132455	0.67690938	0.21
Upper.CI	-0.17389486	0.9656803	0.85132455	1.00000000	0.56480277	0.02
Year	0.02851671	0.6400295	0.67690938	0.56480277	1.00000000	0.29
Numerator_ASD	0.82429404	0.1121787	0.21429644	0.02005452	0.29628163	1.00
Numerator_NonASD	0.99999025	-0.1392238	-0.08080949	-0.17516773	0.02638864	0.82
Proportion	-0.13735462	0.9999677	0.95851437	0.96524017	0.64020778	0.11
Chi_Wilson_Corrected_Lower.CI	-0.08734046	0.9761979	0.99597141	0.88837741	0.67167964	0.19
Chi_Wilson_Corrected_Upper.CI	-0.17380524	0.9798117	0.88384420	0.99561482	0.58775086	0.03

In [346]:
```
# ---------------------------------
# Visualise Correlation Matrix
# ---------------------------------

if(!require(corrplot)){install.packages("corrplot")}
library('corrplot')
```

```
Loading required package: corrplot
corrplot 0.84 loaded
```

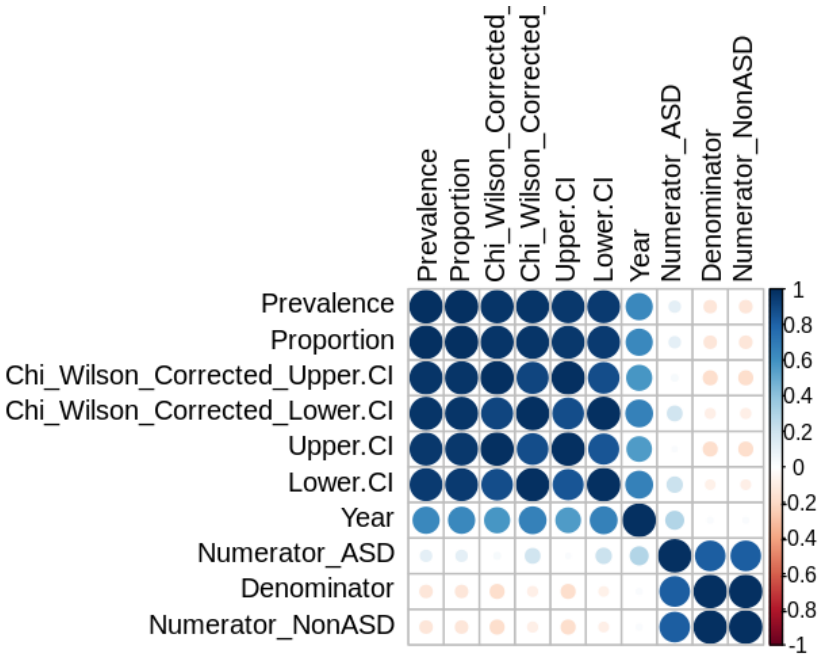
In [347]:
```
# Sort on decreasing correlations with Prevalence
cor_table_sorted <- as.matrix(sort(cor_table[,'Prevalence'], decreasing = TRUE
#
cor_table_sorted
```

Prevalence	1.0000000
Proportion	0.9999677
Chi_Wilson_Corrected_Upper.CI	0.9798117
Chi_Wilson_Corrected_Lower.CI	0.9761979
Upper.CI	0.9656803
Lower.CI	0.9581347
Year	0.6400295
Numerator_ASD	0.1121787
Denominator	-0.1374662
Numerator_NonASD	-0.1392238

In [348]:
```
# Select corelations variables based on threshold:
#cor_var_high <- names(which(apply(cor_table_sorted, 1, function(x) abs(x)>0.2
cor_var_high <- names(which(apply(cor_table_sorted, 1, function(x) abs(x)>0.05
#
cor_var_high
```

'Prevalence' 'Proportion' 'Chi_Wilson_Corrected_Upper.CI' 'Chi_Wilson_Corrected_Lower.CI' 'Upper.CI' 'Lower.CI' 'Year' 'Numerator_ASD' 'Denominator' 'Numerator_NonASD'

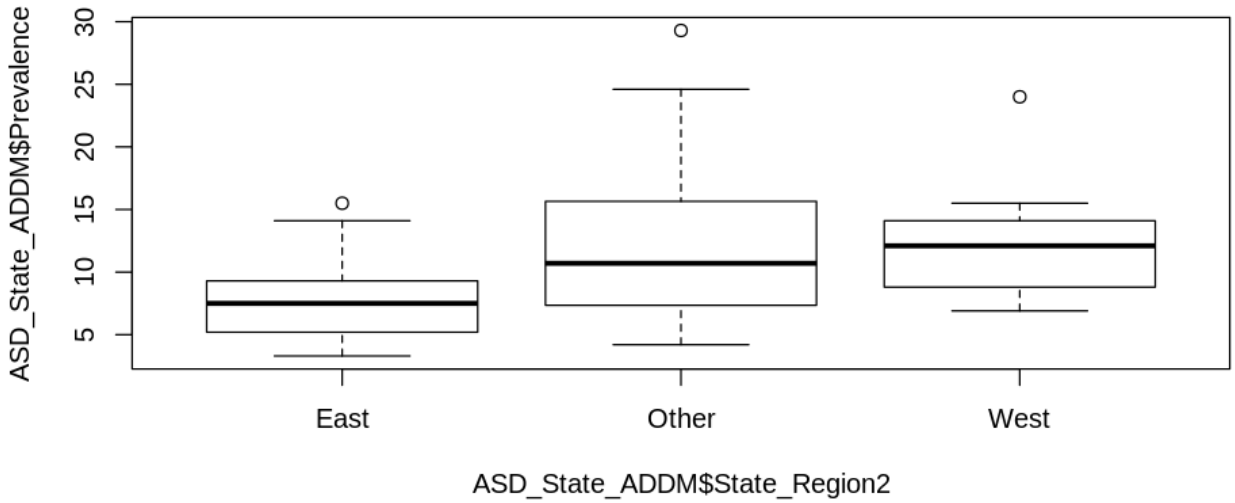
In [349]:
```
# Visualise:
cor_table_plot <- cor_table[cor_var_high, cor_var_high]
# cor_table_plot
#
corrplot(cor_table_plot, tl.col="black", tl.pos = "lt")
```



Hypothesis Test - Mean - Z Test & t.test()

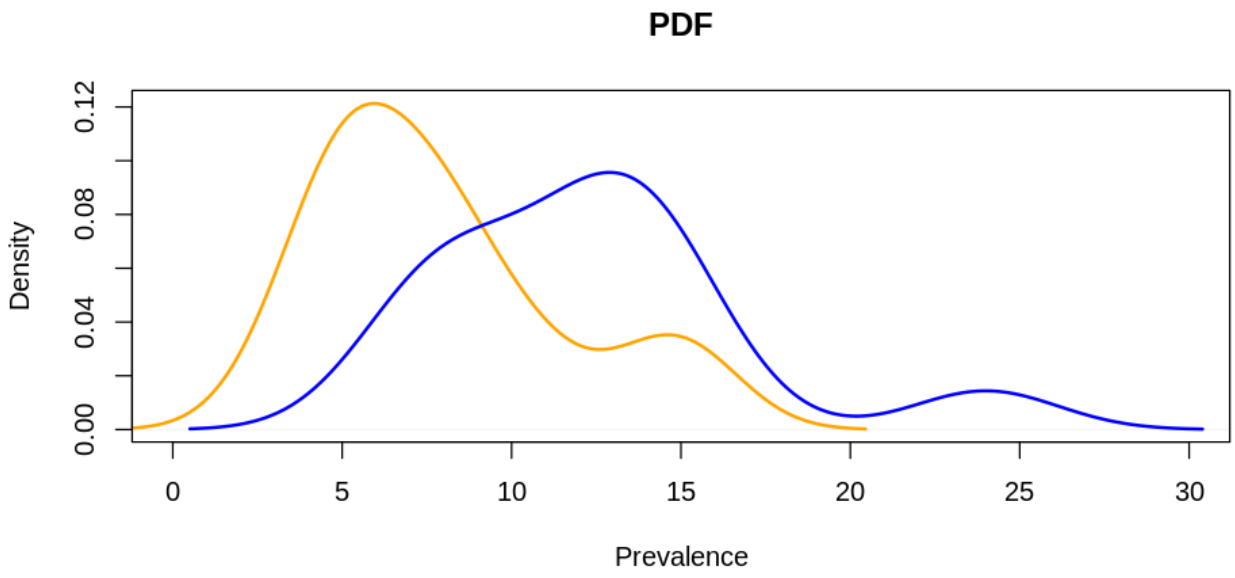
In [350]:

```
#################################################################
# Is there ststistically significant difference of
# ASD Prevelance between East states and West states?
#################################################################

# Aggregate all US. States into three regions: East, West, Other.
ASD_State_ADDM$State_Region2 <- "Other"
ASD_State_ADDM$State_Region2[ASD_State_ADDM$State_Region %in%
                              c("D3 East North Central", "D6 East South Central
ASD_State_ADDM$State_Region2[ASD_State_ADDM$State_Region %in%
                              c("D4 West North Central", "D7 West South Central

boxplot(ASD_State_ADDM$Prevalence ~ ASD_State_ADDM$State_Region2)
```



In [351]:
```
# ---------------------------------
# Hypothesis Test - Mean - Z Test & t.test()
# ---------------------------------

# Create sample 1 of West states
sample_1 = ASD_State_ADDM$Prevalence[ASD_State_ADDM$State_Region2 == "West"]
# Create sample 2 of East states
sample_2 = ASD_State_ADDM$Prevalence[ASD_State_ADDM$State_Region2 == "East"]

# variance test : Equal variance or Unequal variance?
var.test(sample_1, sample_2)

# t test : Equal variance t test
t.test(sample_1, sample_2, var.equal = TRUE)

# Visualise samples:
plot(density(sample_2), col="orange", xlab="Prevalence", main="PDF", lwd=2, lt
lines(density(sample_1), col="blue", lwd=2, lty=1)
```

```
        F test to compare two variances

data:  sample_1 and sample_2
F = 1.4875, num df = 12, denom df = 12, p-value = 0.5019
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.4538834 4.8749547
sample estimates:
ratio of variances
          1.487502


        Two Sample t-test

data:  sample_1 and sample_2
t = 2.7482, df = 24, p-value = 0.01119
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 1.107134 7.785174
sample estimates:
mean of x mean of y
12.323077  7.876923
```



https://www.statsdirect.co.uk/help/Default.htm#parametric_methods/unpaired_t.htm

(https://www.statsdirect.co.uk/help/Default.htm#parametric_methods/unpaired_t.htm)

https://www.bmj.com/about-bmj/resources-readers/publications/statistics-square-one/7-t-tests (https://www.bmj.com/about-bmj/resources-readers/publications/statistics-square-one/7-t-tests)

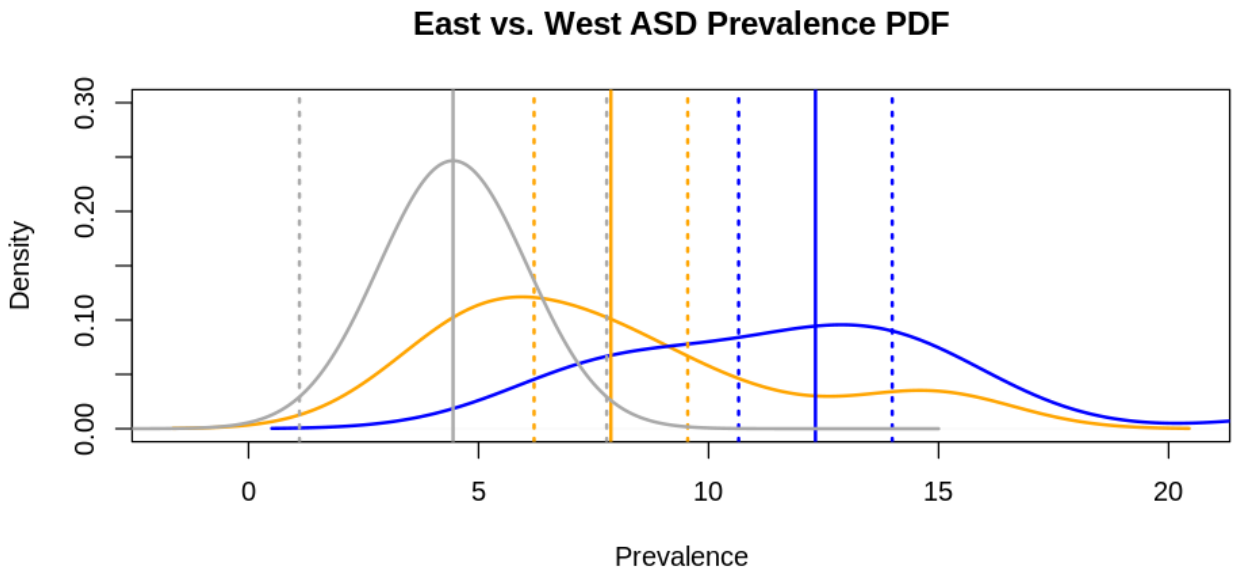
https://ncss-wpengine.netdna-ssl.com/wp-content/themes/ncss/pdf/Procedures/NCSS/Two-Sample_T-Test_from_Means_and_SDs.pdf (https://ncss-wpengine.netdna-ssl.com/wp-content/themes/ncss/pdf/Procedures/NCSS/Two-Sample_T-Test_from_Means_and_SDs.pdf)

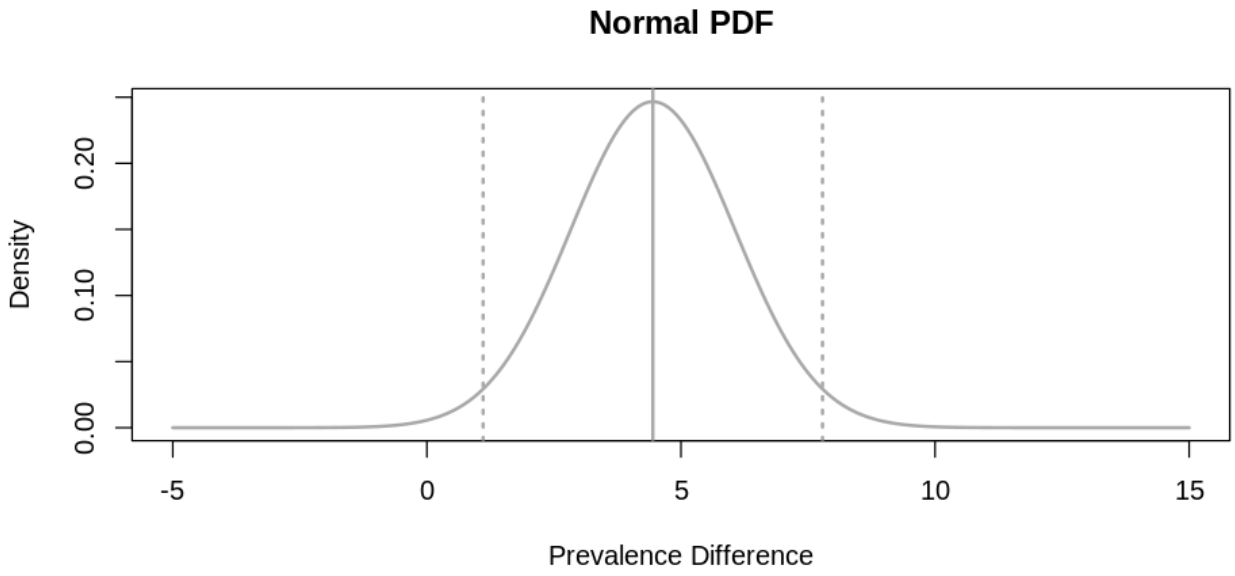
In [352]:

```
#################################################################
# Deep Dive : Equal variance t test
#################################################################

# sample size
cal_n1 = length(sample_1)
cat("\ncal_n1:", cal_n1)
cal_n2 = length(sample_2)
cat("\ncal_n2:", cal_n2)

# degree of freedom
cal_df = cal_n1 + cal_n2 - 2
cat("\ncal_df:", cal_df)

# pooled standard deviation
cal_s = sqrt(((cal_n1 - 1)*sd(sample_1)^2 + (cal_n2 - 1)*sd(sample_2)^2)/(cal_
cat("\ncal_s :", cal_s)

# combined standard error
cal_se = cal_s * sqrt(1/cal_n1 + 1/cal_n2)
cat("\ncal_se:", cal_se)

# t statistic using combined standard error
cal_t = (mean(sample_1)-mean(sample_2)) / cal_se
cat("\ncal_t :", cal_t)

# p-value
cal_p = (1-pt(q = cal_t, df = cal_df)) * 2
cat("\ncal_p :", cal_p)

# 95% CI using "pooled standard deviation"
# sample_1 Upper CI
mn_upper_ci = mean(sample_1) + qt(p = 0.975, df = cal_df)*cal_s/sqrt(cal_n1+ca
cat("\nmn_upper_ci:", mn_upper_ci)
# sample_1 Lower CI
mn_lower_ci = mean(sample_1) - qt(p = 0.975, df = cal_df)*cal_s/sqrt(cal_n1+ca
cat("\nmn_lower_ci:", mn_lower_ci)
# sample_2 Upper CI
fn_upper_ci = mean(sample_2) + qt(p = 0.975, df = cal_df)*cal_s/sqrt(cal_n1+ca
cat("\nfn_upper_ci:", fn_upper_ci)
# sample_2 Lower CI
fn_lower_ci = mean(sample_2) - qt(p = 0.975, df = cal_df)*cal_s/sqrt(cal_n1+ca
cat("\nfn_lower_ci:", fn_lower_ci)
# Difference with 95% CI
cat("\nDifference's Upper CI:", mn_upper_ci - fn_lower_ci)
cat("\nDifference's Lower CI:", mn_lower_ci - fn_upper_ci)

# Difference with 95% CI using combined standard error
# Difference's Upper CI
dif_upper_ci = mean(sample_1)-mean(sample_2) + qt(p = 0.975, df = cal_df)*cal_
cat("\nDifference's Upper CI:", dif_upper_ci)
# Difference's Lower CI
dif_lower_ci = mean(sample_1)-mean(sample_2) - qt(p = 0.975, df = cal_df)*cal_
cat("\nDifference's Lower CI:", dif_lower_ci)

# Visualise CIs: sample_1 & sample_2
plot(density(sample_2), col="orange", xlab="Prevalence", lwd=2, ylim=c(0,0.3),
     main="East vs. West ASD Prevalence PDF")
lines(density(sample_1), col="blue", lwd=2)
# Overlay sample_1 CI
abline(v= mean(sample_1), col="blue", lwd=2, lty=1)
abline(v=mn_upper_ci, col="blue", lwd=2, lty=3)
abline(v=mn_lower_ci, col="blue", lwd=2, lty=3)
# Overlay sample_2 CI
abline(v= mean(sample_2), col="orange", lwd=2, lty=1)
```

```r
abline(v=fn_upper_ci, col="orange", lwd=2, lty=3)
abline(v=fn_lower_ci, col="orange", lwd=2, lty=3)

# Overlay Difference's CI
abline(v=mean(sample_1)-mean(sample_2), col="darkgrey", lwd=2, lty=1)
abline(v=dif_upper_ci, col="darkgrey", lwd=2, lty=3)
abline(v=dif_lower_ci, col="darkgrey", lwd=2, lty=3)
cal_xseq <- seq(-5,15,0.01)
cal_pdf <- dnorm(x = cal_xseq, mean = mean(sample_1)-mean(sample_2), sd = cal_
lines(cal_xseq, cal_pdf, col="darkgrey", type="l",lwd=2)

# Visualise Difference's CIs alone:
plot(cal_xseq, cal_pdf, col="darkgrey", xlab="Prevalence Difference", ylab="De
     main="Normal PDF")
# Overlay Difference's CI
abline(v=mean(sample_1)-mean(sample_2), col="darkgrey", lwd=2, lty=1)
abline(v=dif_upper_ci, col="darkgrey", lwd=2, lty=3)
abline(v=dif_lower_ci, col="darkgrey", lwd=2, lty=3)
```

```
cal_n1: 13
cal_n2: 13
cal_df: 24
cal_s : 4.124652
cal_se: 1.617822
cal_t : 2.748235
cal_p : 0.01119334
mn_upper_ci: 13.99259
mn_lower_ci: 10.65357
fn_upper_ci: 9.546433
fn_lower_ci: 6.207413
Difference's Upper CI: 7.785174
Difference's Lower CI: 1.107134
Difference's Upper CI: 7.785174
Difference's Lower CI: 1.107134
```





Hypothesis Test - Proportion - prop.test()

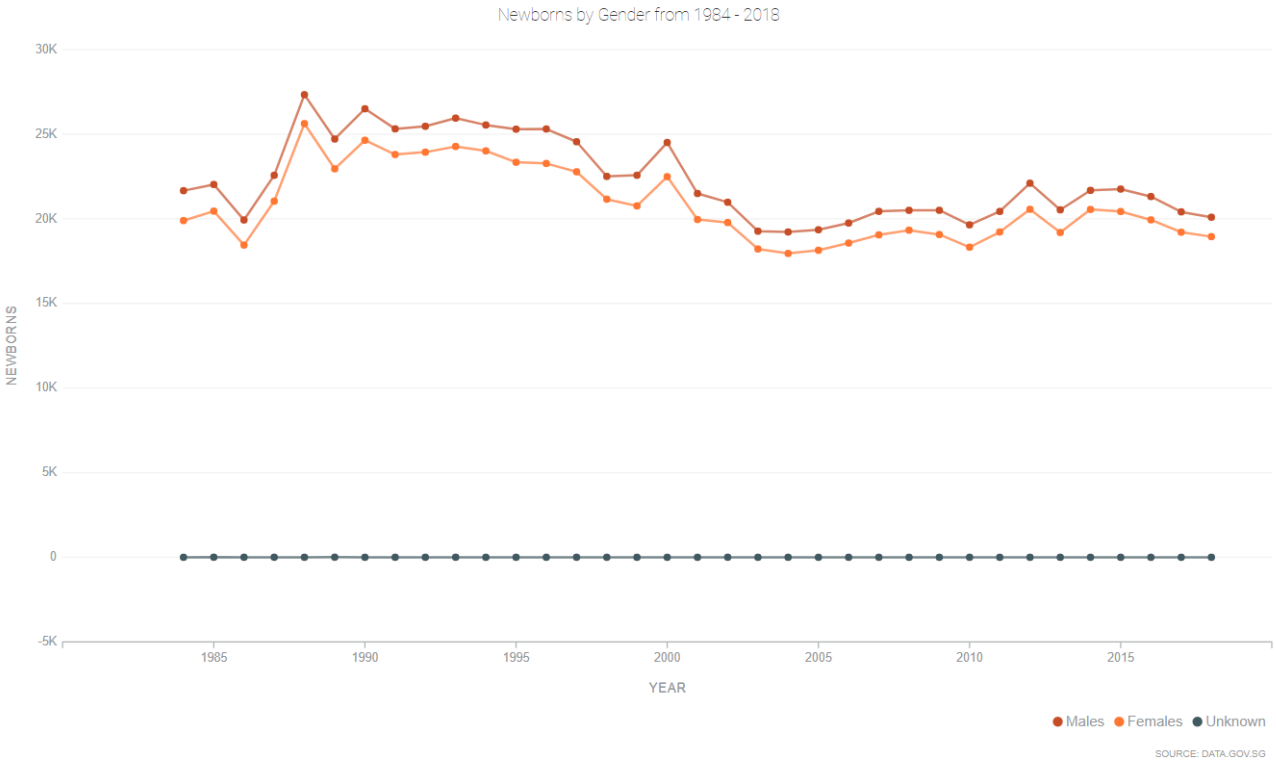
https://en.wikipedia.org/wiki/Binomial_proportion_confidence_interval (https://en.wikipedia.org/wiki/Binomial_proportion_confidence_interval)

In [353]:
```
# ---------------------------------
# Hypothesis Test - Proportion - prop.test()
# ---------------------------------

# Two group hypothesis test : proportions (Prevalvence) among two US. States
prop.test(ASD[1:2], Children[1:2])

# Multiple group hypothesis test : proportions (Prevalvence) among all US. Sta
prop.test(ASD, Children)
```

```
        2-sample test for equality of proportions with continuity correction

data:  ASD[1:2] out of Children[1:2]
X-squared = 10.922, df = 1, p-value = 0.0009503
alternative hypothesis: two.sided
95 percent confidence interval:
 -0.0016149661 -0.0003828408
sample estimates:
     prop 1      prop 2
0.009100327 0.010099231


        50-sample test for equality of proportions without continuity
        correction

data:  ASD out of Children
X-squared = 28593, df = 49, p-value < 2.2e-16
alternative hypothesis: two.sided
sample estimates:
     prop 1      prop 2      prop 3      prop 4      prop 5      prop 6
0.009100327 0.010099231 0.010400284 0.009499466 0.013900007 0.007299433
     prop 7      prop 8      prop 9     prop 10     prop 11     prop 12
0.015399168 0.011098717 0.011896793 0.012099872 0.010300278 0.008299863
    prop 13     prop 14     prop 15     prop 16     prop 17     prop 18
0.009498530 0.011000237 0.014200361 0.001399357 0.008200786 0.009299486
    prop 19     prop 20     prop 21     prop 22     prop 23     prop 24
0.006900015 0.016696937 0.011900547 0.017399589 0.011199751 0.019400428
    prop 25     prop 26     prop 27     prop 28     prop 29     prop 30
0.009499812 0.012399961 0.005500221 0.010800730 0.012698882 0.014097784
    prop 31     prop 32     prop 33     prop 34     prop 35     prop 36
0.014100222 0.008501570 0.012999924 0.011199812 0.009796534 0.012499686
    prop 37     prop 38     prop 39     prop 40     prop 41     prop 42
0.008899480 0.015100037 0.016899883 0.015201438 0.009599935 0.008997355
    prop 43     prop 44     prop 45     prop 46     prop 47     prop 48
0.009500416 0.010199940 0.008699269 0.012103956 0.014200156 0.011200227
    prop 49     prop 50
0.008601179 0.009299132
```

Hypothesis Test - Case Study: Understanding the newborn gender differences in Singapore

https://data.gov.sg/dataset/newborns-by-gender (https://data.gov.sg/dataset/newborns-by-gender)



Year	Gender	Newborns (No. of Newborns)
2018	Males	20,093
2018	Females	18,945
2018	Unknown	1
2017	Males	20,408
2017	Females	19,207
2017	Unknown	0
2016	Males	21,315
2016	Females	19,936
2016	Unknown	0
2015	Males	21,755
2015	Females	20,430
2015	Unknown	0
2014	Males	21,679
2014	Females	20,552
2014	Unknown	1

Showing 1 to 15 of 105 records

1 2 3 4 5 6 7 »

SOURCE: DATA.GOV.SG

Above tabular CSV data from Data.gov.sg https://data.gov.sg/dataset/e55726f8-6d91-4cad-8e40-d4aa10ec7877/download (https://data.gov.sg/dataset/e55726f8-6d91-4cad-8e40-

d4aa10ec7877/download)

Read in CSV data, storing as R dataframe

In [354]:
```
df_newborns <- read.csv("../reference/newborns-by-gender/newborns-by-gender-fr
head(df_newborns)
tail(df_newborns)
```

year	gender	newborns
1984	Males	21661
1984	Females	19894
1984	Unknown	1
1985	Males	22027
1985	Females	20452
1985	Unknown	5

	year	gender	newborns
100	2017	Males	20408
101	2017	Females	19207
102	2017	Unknown	0
103	2018	Males	20093
104	2018	Females	18945
105	2018	Unknown	1

In [355]:
```
# Example of filtering data
subset(df_newborns, year == 2018, select = c(year, gender, newborns))
```

	year	gender	newborns
103	2018	Males	20093
104	2018	Females	18945
105	2018	Unknown	1

In [356]:
```
# year 2018 Male newborn number
subset(df_newborns, year == 2018 & gender == "Males")$newborns
```

20093

In [357]:
```
# year 2018 Female newborn number
subset(df_newborns, year == 2018 & gender == "Females")$newborns
```

18945

Is the Male newborn proportion similar (statistically equal) to Female newborn proportion, expected as a natural 50%-50%?

One sample proportion test:

In [358]:
```
# year 2018 Male newborn proportion vs. expected 50% ratio (null probability 0
mn = subset(df_newborns, year == 2018 & gender == "Males")$newborns # Male new
fn = subset(df_newborns, year == 2018 & gender == "Females")$newborns # Female
total = mn + fn

prop.test(x = mn, n = total)
# prop.test(x = mn, n = total, alternative = "greater")
```

```
        1-sample proportions test with continuity correction

data:  mn out of total, null probability 0.5
X-squared = 33.701, df = 1, p-value = 6.428e-09
alternative hypothesis: true p is not equal to 0.5
95 percent confidence interval:
 0.5097318 0.5196725
sample estimates:
        p
0.5147036
```

Is there a (statistically significant) change of Male newborn ratio now and then, i.e. 2018 vs. 1984?

Two sample proportion test:

In [359]:
```
subset(df_newborns[order(-df_newborns$year),], year %in% c(2018, 1984) & gende
```

	year	gender	newborns
103	2018	Males	20093
1	1984	Males	21661

In [360]:
```
subset(df_newborns[order(-df_newborns$year),], year %in% c(2018, 1984) & gende
```

	year	gender	newborns
104	2018	Females	18945
2	1984	Females	19894

In [361]:
```
# year 2018 Male newborn proportion vs. year 1984 Male newborn proportion
mn = subset(df_newborns[order(-df_newborns$year),], year %in% c(2018, 1984) &
fn = subset(df_newborns[order(-df_newborns$year),], year %in% c(2018, 1984) &
total = mn + fn

prop.test(x = mn, n = total)
# prop.test(x = mn, n = total, alternative = "less")
```

```
        2-sample test for equality of proportions with continuity correction

data:  mn out of total
X-squared = 3.4404, df = 1, p-value = 0.06362
alternative hypothesis: two.sided
95 percent confidence interval:
 -0.0134849792  0.0003702646
sample estimates:
   prop 1    prop 2
0.5147036 0.5212610
```

Is there a (statistically significant) difference between average number of Male newborns and average number of Female newborns?

Two sample mean test (t test):

In [362]:
```
mn_all = subset(df_newborns[order(-df_newborns$year),], gender == "Males")
dim(mn_all)
head(mn_all)
```

35 3

	year	gender	newborns
103	2018	Males	20093
100	2017	Males	20408
97	2016	Males	21315
94	2015	Males	21755
91	2014	Males	21679
88	2013	Males	20528

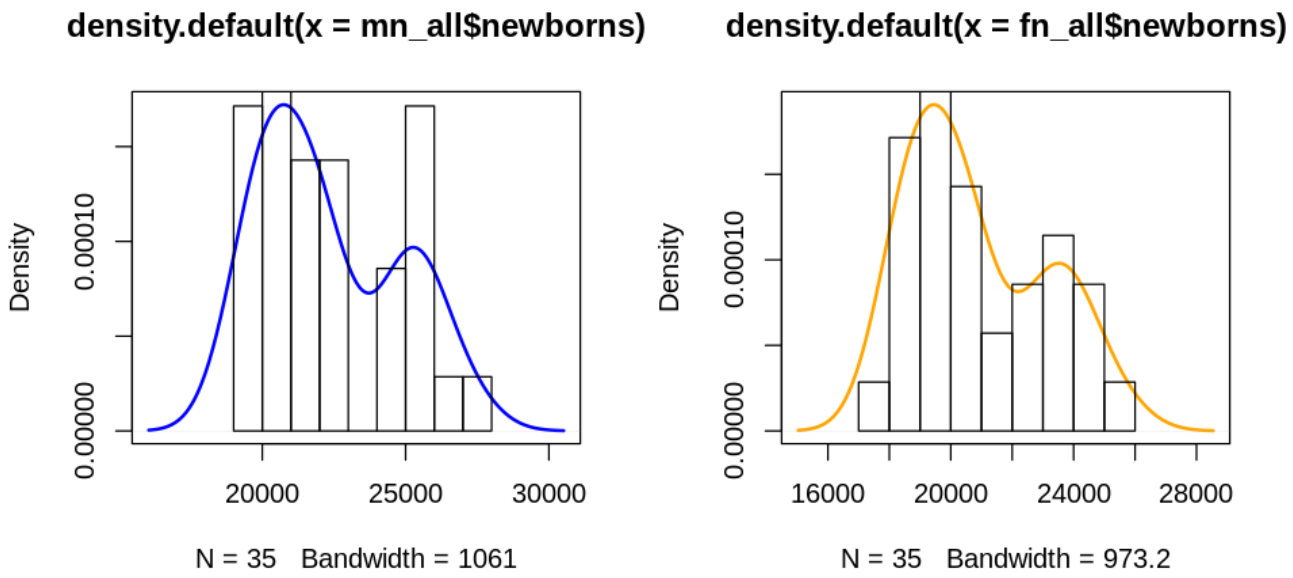
In [363]:
```
fn_all = subset(df_newborns[order(-df_newborns$year),], gender == "Females")
dim(fn_all)
head(fn_all)
```

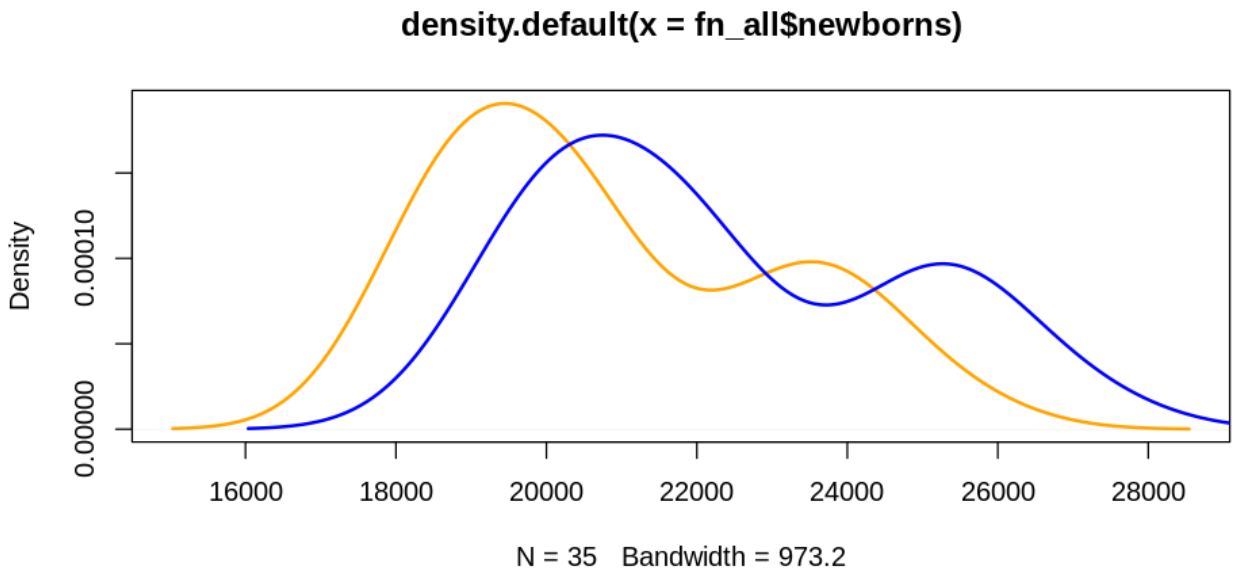
35 3

	year	gender	newborns
104	2018	Females	18945
101	2017	Females	19207
98	2016	Females	19936
95	2015	Females	20430
92	2014	Females	20552
89	2013	Females	19191

In [364]:
```
cat("\nMale   average newborns:", mean(mn_all$newborns)) # Male average
cat("\nFemale average newborns:", mean(fn_all$newborns)) # Female average

# Adjust in-line plot size to M x N
options(repr.plot.width=8, repr.plot.height=4)

par(mfrow=c(1, 2))
# Male
plot(density(mn_all$newborns), col="blue", lwd=2)
hist(mn_all$newborns, probability = T, add = T)
# Female
plot(density(fn_all$newborns), col="orange", lwd=2)
hist(fn_all$newborns, probability = T, add = T)

par(mfrow=c(1, 1))

plot(density(fn_all$newborns), col="orange", lwd=2)
lines(density(mn_all$newborns), col="blue", lwd=2)
```

```
Male   average newborns: 22319.94
Female average newborns: 20836.89
```





In [365]:
```
# t test with equal variance:
t.test(mn_all$newborns, fn_all$newborns, paired = FALSE, var.equal = TRUE)
# t.test(mn_all$newborns, fn_all$newborns, paired = FALSE, var.equal = TRUE, a
```

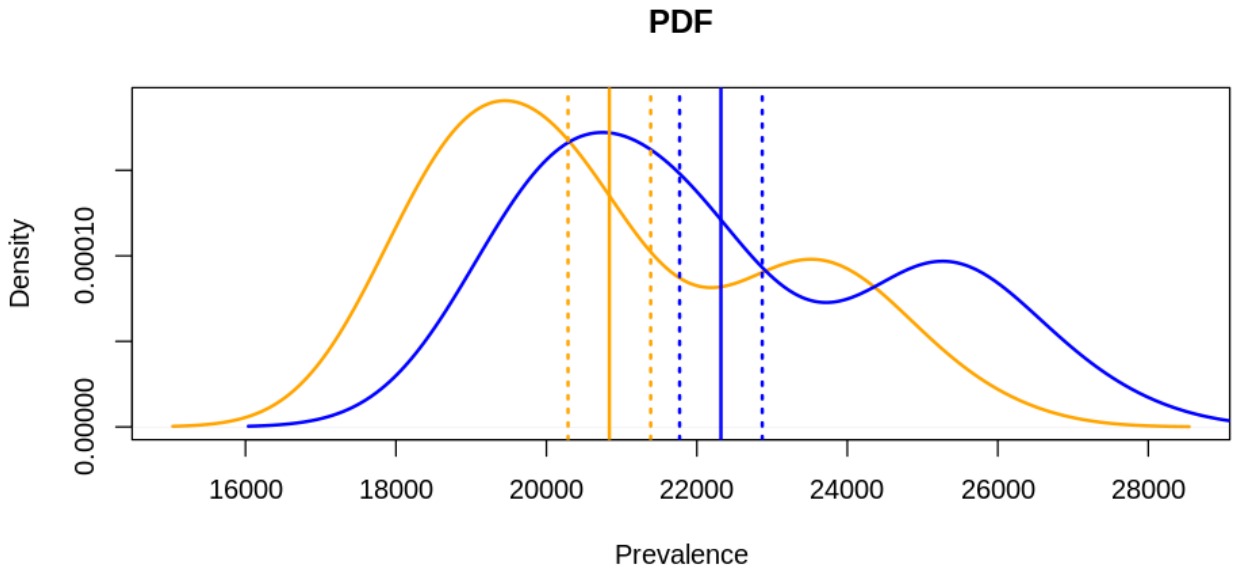
```
        Two Sample t-test

data:  mn_all$newborns and fn_all$newborns
t = 2.6935, df = 68, p-value = 0.008898
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
  384.3287 2581.7855
sample estimates:
mean of x mean of y
 22319.94  20836.89
```

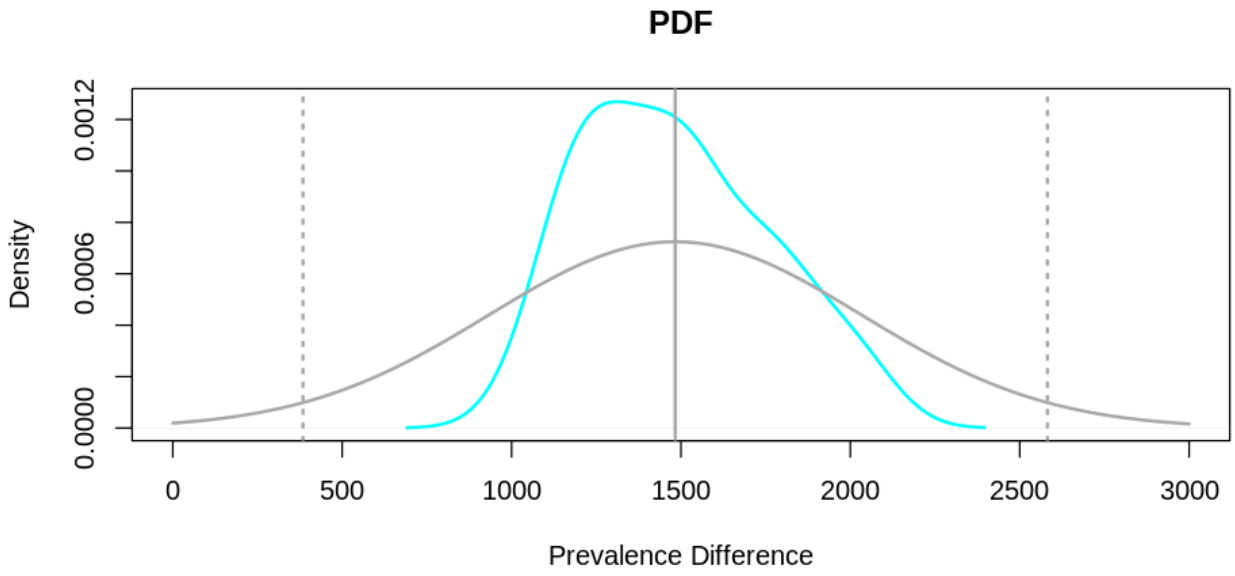
In [366]:

```
#################################################################
# Deep Dive : Equal variance t test
#################################################################

# sample size
cal_n1 = length(mn_all$newborns)
cat("\ncal_n1:", cal_n1)
cal_n2 = length(fn_all$newborns)
cat("\ncal_n2:", cal_n2)

# degree of freedom
cal_df = cal_n1 + cal_n2 - 2
cat("\ncal_df:", cal_df)

# pooled standard deviation
cal_s = sqrt(((cal_n1 - 1)*sd(mn_all$newborns)^2 + (cal_n2 - 1)*sd(fn_all$newb
cat("\ncal_s :", cal_s)

# combined standard error
cal_se = cal_s*sqrt(1/cal_n1 + 1/cal_n2)
cat("\ncal_se:", cal_se)

# t statistic using combined standard error
cal_t = (mean(mn_all$newborns)-mean(fn_all$newborns))/cal_se
cat("\ncal_t :", cal_t)

# p-value
cal_p = (1-pt(q = cal_t, df = cal_df))*2
cat("\ncal_p :", cal_p)

# 95% CI using "pooled standard deviation"
# Male Upper CI
mn_upper_ci = mean(mn_all$newborns) + qt(p = 0.975, df = cal_df)*cal_s/sqrt(ca
cat("\nmn_upper_ci:", mn_upper_ci)
# Male Lower CI
mn_lower_ci = mean(mn_all$newborns) - qt(p = 0.975, df = cal_df)*cal_s/sqrt(ca
cat("\nmn_lower_ci:", mn_lower_ci)
# Female Upper CI
fn_upper_ci = mean(fn_all$newborns) + qt(p = 0.975, df = cal_df)*cal_s/sqrt(ca
cat("\nfn_upper_ci:", fn_upper_ci)
# Female Lower CI
fn_lower_ci = mean(fn_all$newborns) - qt(p = 0.975, df = cal_df)*cal_s/sqrt(ca
cat("\nfn_lower_ci:", fn_lower_ci)
# Difference with 95% CI
cat("\nDifference's Upper CI:", mn_upper_ci - fn_lower_ci)
cat("\nDifference's Lower CI:", mn_lower_ci - fn_upper_ci)

# Difference with 95% CI using combined standard error
# Difference's Upper CI
dif_upper_ci = mean(mn_all$newborns)-mean(fn_all$newborns) + qt(p = 0.975, df
cat("\nDifference's Upper CI:", dif_upper_ci)
# Difference's Lower CI
dif_lower_ci = mean(mn_all$newborns)-mean(fn_all$newborns) - qt(p = 0.975, df
cat("\nDifference's Lower CI:", dif_lower_ci)

# Visualise CIs: Male & Female
plot(density(fn_all$newborns), col="orange", xlab="Prevalence", main="PDF", lw
lines(density(mn_all$newborns), col="blue", lwd=2)
# Overlay Male CI
abline(v= mean(mn_all$newborns), col="blue", lwd=2, lty=1)
abline(v=mn_upper_ci, col="blue", lwd=2, lty=3)
abline(v=mn_lower_ci, col="blue", lwd=2, lty=3)
# Overlay Female CI
abline(v= mean(fn_all$newborns), col="orange", lwd=2, lty=1)
abline(v=fn_upper_ci, col="orange", lwd=2, lty=3)
```

```r
abline(v=fn_lower_ci, col="orange", lwd=2, lty=3)

# Visualise CIs: Difference between Male and Female
plot(density(mn_all$newborns-fn_all$newborns), col="cyan", xlab="Prevalence Di
     lwd=2, lty=1, xlim=c(0,3000))
# Overlay Difference's CI
abline(v=mean(mn_all$newborns)-mean(fn_all$newborns), col="darkgrey", lwd=2, l
abline(v=dif_upper_ci, col="darkgrey", lwd=2, lty=3)
abline(v=dif_lower_ci, col="darkgrey", lwd=2, lty=3)
cal_xseq <- seq(0,3000,0.01)
cal_pdf <- dnorm(x = cal_xseq, mean = mean(mn_all$newborns)-mean(fn_all$newbor
lines(cal_xseq, cal_pdf, col="darkgrey", type="l",lwd=2)
```

```
cal_n1: 35
cal_n2: 35
cal_df: 68
cal_s : 2303.374
cal_se: 550.6116
cal_t : 2.693472
cal_p : 0.00889784
mn_upper_ci: 22869.31
mn_lower_ci: 21770.58
fn_upper_ci: 21386.25
fn_lower_ci: 20287.52
Difference's Upper CI: 2581.786
Difference's Lower CI: 384.3287
Difference's Upper CI: 2581.786
Difference's Lower CI: 384.3287
```





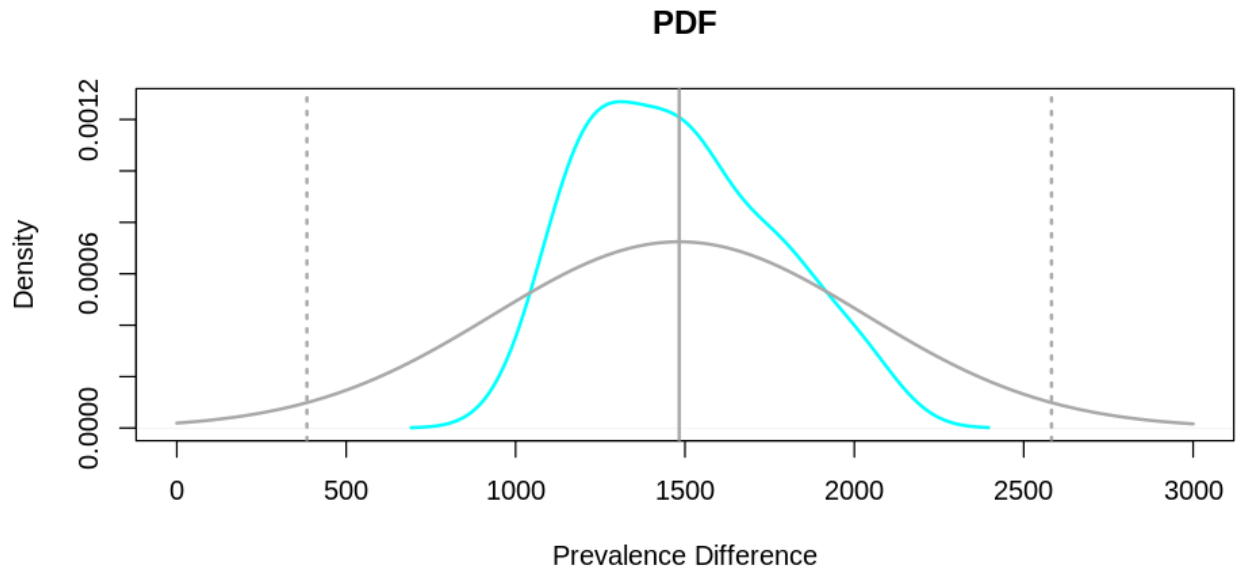
In [367]:
```
# t test with unequal variances
t.test(mn_all$newborns, fn_all$newborns, paired = FALSE)
# t.test(mn_all$newborns, fn_all$newborns, paired = FALSE, alternative = "grea
```

```
        Welch Two Sample t-test

data:  mn_all$newborns and fn_all$newborns
t = 2.6935, df = 67.497, p-value = 0.008912
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
  384.1804 2581.9339
sample estimates:
mean of x mean of y
 22319.94  20836.89
```

In [368]:

```
#################################################################
# Deep Dive : Unequal variance t test
#################################################################

# sample size
cal_n1 = length(mn_all$newborns)
cat("\ncal_n1:", cal_n1)
cal_n2 = length(fn_all$newborns)
cat("\ncal_n2:", cal_n2)

# degree of freedom
# cal_df = cal_n1 + cal_n2 - 2 # <-- This is df of equal variances
cal_df = ( sd(mn_all$newborns)^2/cal_n1 + sd(fn_all$newborns)^2/cal_n2 )^2 /
    ((sd(mn_all$newborns)^2/cal_n1)^2/(cal_n1 - 1) + (sd(fn_all$newborns)^2/ca
cat("\ncal_df:", cal_df)

# combined standard error: Welch's denominator (It is NOT a pooled standard de
cal_se = sqrt( sd(mn_all$newborns)^2/cal_n1 + sd(fn_all$newborns)^2/cal_n2 )
cat("\ncal_se:", cal_se)

# t statistic using combined standard error: Welch's denominator
cal_t = (mean(mn_all$newborns)-mean(fn_all$newborns))/cal_se
cat("\ncal_t :", cal_t)

# p-value
cal_p = (1-pt(q = cal_t, df = cal_df))*2
cat("\ncal_p :", cal_p)

# Difference with 95% CI using combined standard error
# Difference's Upper CI
dif_upper_ci = mean(mn_all$newborns)-mean(fn_all$newborns) + qt(p = 0.975, df
cat("\nDifference's Upper CI:", dif_upper_ci)
# Difference's Lower CI
dif_lower_ci = mean(mn_all$newborns)-mean(fn_all$newborns) - qt(p = 0.975, df
cat("\nDifference's Lower CI:", dif_lower_ci)

# Visualise CIs:
plot(density(mn_all$newborns-fn_all$newborns), col="cyan", xlab="Prevalence Di
     lwd=2, lty=1, xlim=c(0,3000))
# Overlay Difference's CI
abline(v=mean(mn_all$newborns)-mean(fn_all$newborns), col="darkgrey", lwd=2, l
abline(v=dif_upper_ci, col="darkgrey", lwd=2, lty=3)
abline(v=dif_lower_ci, col="darkgrey", lwd=2, lty=3)
cal_xseq <- seq(0,3000,0.01)
cal_pdf <- dnorm(x = cal_xseq, mean = mean(mn_all$newborns)-mean(fn_all$newbor
lines(cal_xseq, cal_pdf, col="darkgrey", type="l",lwd=2)
```

```
cal_n1: 35
cal_n2: 35
cal_df: 67.49676
cal_se: 550.6116
cal_t : 2.693472
cal_p : 0.008912138
Difference's Upper CI: 2581.934
Difference's Lower CI: 384.1804
```



Is the difference(s) between number of Male newborns and number of Female newborns stable (non-random) over the years?

Two sample mean test (paired t test):

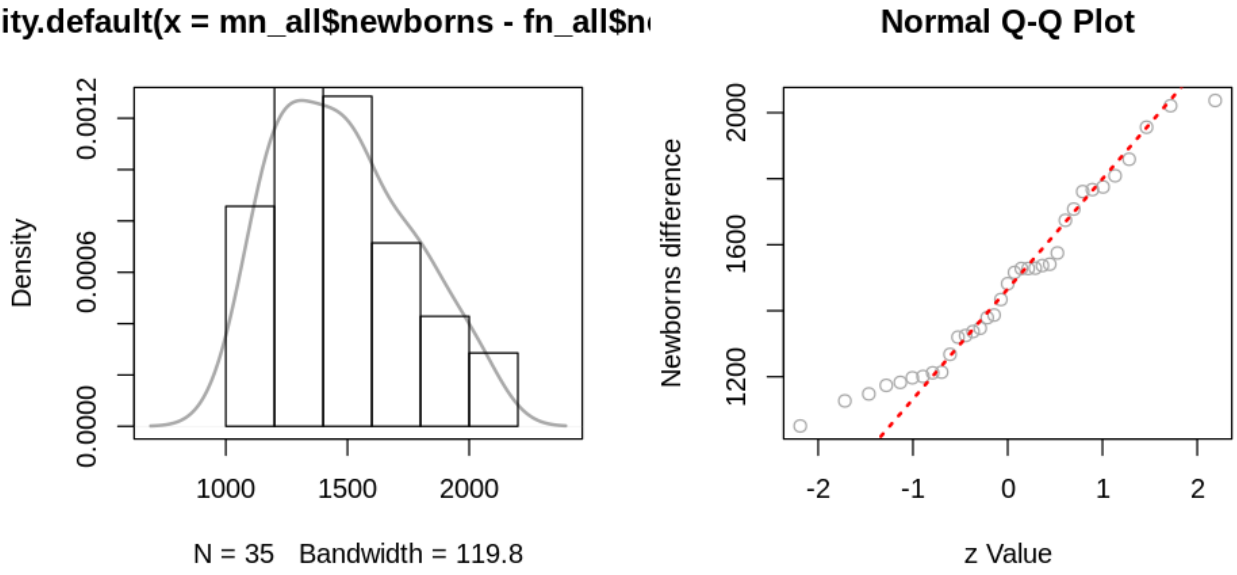
In [369]:
```
t.test(mn_all$newborns, fn_all$newborns, paired = TRUE)
# t.test(mn_all$newborns, fn_all$newborns, paired = TRUE, alternative = "great
```

```
        Paired t-test

data:  mn_all$newborns and fn_all$newborns
t = 32.368, df = 34, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 1389.942 1576.172
sample estimates:
mean of the differences
               1483.057
```

[Tips] Paired t test essentially evaluates the difference of two samples, assuming 'the difference' forming a t/normal distribution.

In [370]:
```
# Difference between Male and Female newborns

# Normality test:
shapiro.test(mn_all$newborns - fn_all$newborns)

# Visualise:
par(mfrow=c(1, 2))
plot(density(mn_all$newborns - fn_all$newborns), col="darkgrey", lwd=2)
hist(mn_all$newborns - fn_all$newborns, probability = T, add = T)

qqnorm(mn_all$newborns - fn_all$newborns, col="darkgrey",
       xlab="z Value", ylab="Newborns difference")
qqline(mn_all$newborns - fn_all$newborns, col="red", lwd=2, lty=3)
par(mfrow=c(1, 1))
```

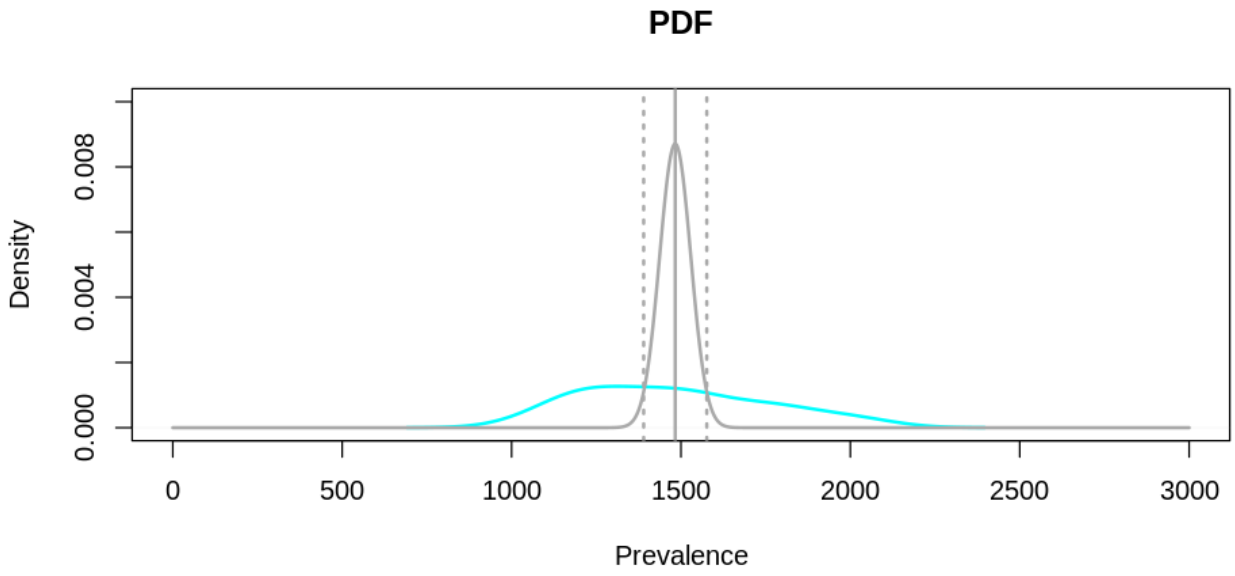
```
        Shapiro-Wilk normality test

data:  mn_all$newborns - fn_all$newborns
W = 0.9522, p-value = 0.1324
```



In [371]:

```
#############################################################
# Deep Dive : Paired t test
#############################################################

# sample size
cal_n = length(mn_all$newborns - fn_all$newborns)
cat("\ncal_n :", cal_n)

# degree of freedom
cal_df = cal_n - 1
cat("\ncal_df:", cal_df)

# standard deviation
cal_s = sd(mn_all$newborns - fn_all$newborns)
cat("\ncal_s :", cal_s)

# standard error
cal_se = cal_s / sqrt(cal_n)
cat("\ncal_se:", cal_se)

# t statistic calculated using standard error:
cal_t = mean(mn_all$newborns-fn_all$newborns)/cal_se
cat("\ncal_t :", cal_t)

# p-value
cal_p = (1-pt(q = cal_t, df = cal_df))*2
cat("\ncal_p :", cal_p)

# 95% CI using "standard deviation"
# Paired Difference's Upper CI
dif_upper_ci = mean(mn_all$newborns-fn_all$newborns) + qt(p = 0.975, df = cal_
cat("\ndif_upper_ci:", dif_upper_ci)
# Paired Difference's Lower CI
dif_lower_ci = mean(mn_all$newborns-fn_all$newborns) - qt(p = 0.975, df = cal_
cat("\ndif_lower_ci:", dif_lower_ci)

# Visualise CIs:
plot(density(mn_all$newborns-fn_all$newborns), col="cyan", xlab="Prevalence",
     lwd=2, lty=1, xlim=c(0,3000), ylim=c(0,0.01))
# Overlay CI
abline(v= mean(mn_all$newborns-fn_all$newborns), col="darkgrey", lwd=2, lty=1)
abline(v=dif_upper_ci, col="darkgrey", lwd=2, lty=3)
abline(v=dif_lower_ci, col="darkgrey", lwd=2, lty=3)
cal_xseq <- seq(0,3000,0.01)
cal_pdf <- dnorm(x = cal_xseq, mean = mean(mn_all$newborns-fn_all$newborns), s
lines(cal_xseq, cal_pdf, col="darkgrey", type="l",lwd=2)

cat("\nPaired Difference's Upper CI:", dif_upper_ci)
cat("\nPaired Difference's Lower CI:", dif_lower_ci)
```

```
cal_n : 35
cal_df: 34
cal_s : 271.0677
cal_se: 45.81881
cal_t : 32.36787
cal_p : 0
dif_upper_ci: 1576.172
dif_lower_ci: 1389.942
Paired Difference's Upper CI: 1576.172
Paired Difference's Lower CI: 1389.942
```

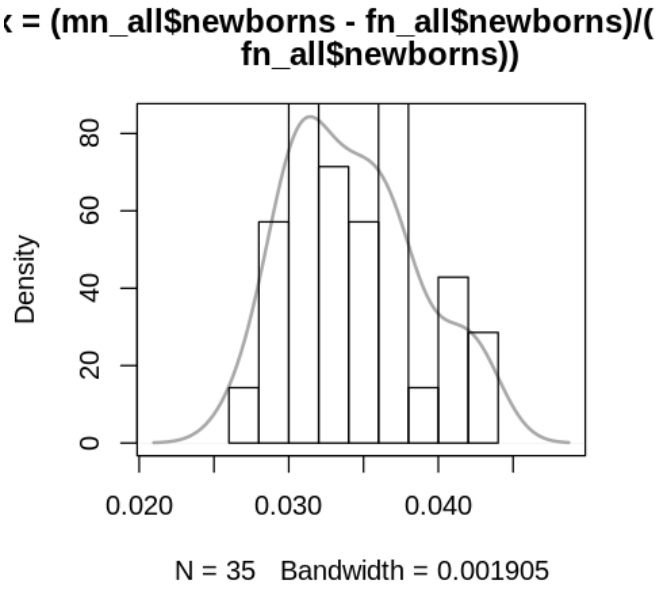


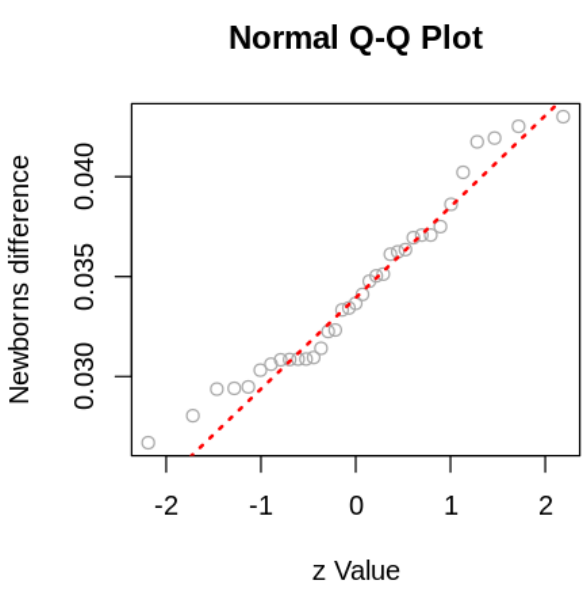
Is the difference(s) between Male newborns ratio/proportion and Female newborns ratio/proportion stable (non-random) over the years?

Two sample mean test (paired t test):

In [372]:
```
# Proportion difference between Male and Female newborns

# Uncomment below to display Male & Female newborns ratio/proportion over year
# mn_all$newborns/(mn_all$newborns+fn_all$newborns)
# fn_all$newborns/(mn_all$newborns+fn_all$newborns)

t.test(mn_all$newborns/(mn_all$newborns+fn_all$newborns),
       fn_all$newborns/(mn_all$newborns+fn_all$newborns), paired = TRUE)
t.test(mn_all$newborns/(mn_all$newborns+fn_all$newborns),
       fn_all$newborns/(mn_all$newborns+fn_all$newborns), paired = TRUE, alter

# Normality test:
shapiro.test((mn_all$newborns - fn_all$newborns)/(mn_all$newborns+fn_all$newbo

# Visualise:
par(mfrow=c(1, 2))
plot(density((mn_all$newborns - fn_all$newborns)/(mn_all$newborns+fn_all$newbo
hist((mn_all$newborns - fn_all$newborns)/(mn_all$newborns+fn_all$newborns), pr

qqnorm((mn_all$newborns - fn_all$newborns)/(mn_all$newborns+fn_all$newborns),
       xlab="z Value", ylab="Newborns difference")
qqline((mn_all$newborns - fn_all$newborns)/(mn_all$newborns+fn_all$newborns),
par(mfrow=c(1, 1))
```

```
        Paired t-test

data:  mn_all$newborns/(mn_all$newborns + fn_all$newborns) and fn_all$newborn
s/(mn_all$newborns + fn_all$newborns)
t = 47.032, df = 34, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 0.03277953 0.03574025
sample estimates:
mean of the differences
             0.03425989


        Paired t-test

data:  mn_all$newborns/(mn_all$newborns + fn_all$newborns) and fn_all$newborn
s/(mn_all$newborns + fn_all$newborns)
t = 47.032, df = 34, p-value < 2.2e-16
alternative hypothesis: true difference in means is greater than 0
95 percent confidence interval:
 0.03302816        Inf
sample estimates:
mean of the differences
             0.03425989


        Shapiro-Wilk normality test

data:  (mn_all$newborns - fn_all$newborns)/(mn_all$newborns + fn_all$newborn
s)
W = 0.95524, p-value = 0.1641
```





(https://github.com/dd-consulting)