WayneLi7 html 7c776e2 Oct 21, 2017

1 contributor

306 lines (262 sloc) | 777 KB Raw Blame History

```
<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta charset="utf-8" />
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<meta name="generator" content="pandoc" />




<title>post01-Wei-Li</title>


<meta name="viewport" content="width=device-width, initial-scale=1" />

<script src="data:application/x-javascript;base64,LyohCiAqIEJvb3RzdHJhcCB2My4zLjUgKGh0dHA6Ly9nZXRib290c3RyYXAuY29tKQogKiBDb3B5c
<script src="data:application/x-javascript;base64,LyoqCiogQHByZXNlcnZlIEhUTw1IFNoaXYgMy43LjIgfCBAYWZhcmthcyBAamRhbHRvbiBAam9uX
<script src="data:application/x-javascript;base64,LyohIFJlc3BvbmQuanMgdjEuNC4yOiBtaW4vbWF4LXdpZHRoIG1lZGlhIHF1ZXJ5IHBvbHlmaWxsI
<script src="data:application/x-javascript;base64,CgovKioKICogalF1ZXJ5IFBsdWdpbjogU3RpY2t5IFRhYnMKICoKICogQGF1dGhvciBBaWRhbiBMa
<link href="data:text/css;charset=utf-8,pre%20%2Eoperator%2C%0Apre%20%2Eparen%20%7B%0Acolor%3A%20rgb%28104%2C%20118%2C%20135%29
<script src="data:application/x-javascript;base64,dmFyIGhsanM9bmV3IGZ1bmN0aW9uKCl7ZnVuY3Rpb24gbShwKXtyZXR1cm4gcC5yZXBsYWNlKC8mL

<style type="text/css">code{white-space: pre;}</style>
<style type="text/css">
  pre:not([class]) {
    background-color: white;
  }
</style>
<script type="text/javascript">
if (window.hljs && document.readyState && document.readyState === "complete") {
   window.setTimeout(function() {
      hljs.initHighlighting();
   }, 0);
}
</script>



<style type="text/css">
h1 {
  font-size: 34px;
}
h1.title {
  font-size: 38px;
}
h2 {
  font-size: 30px;
}
h3 {
  font-size: 24px;
}
h4 {
  font-size: 18px;
}
h5 {
  font-size: 16px;
}
h6 {
```

```
  font-size: 12px;
}
.table th:not([align]) {
  text-align: left;
}
</style>


</head>

<body>

<style type="text/css">
.main-container {
  max-width: 940px;
  margin-left: auto;
  margin-right: auto;
}
code {
  color: inherit;
  background-color: rgba(0, 0, 0, 0.04);
}
img {
  max-width:100%;
  height: auto;
}
.tabbed-pane {
  padding-top: 12px;
}
button.code-folding-btn:focus {
  outline: none;
}
</style>



<div class="container-fluid main-container">

<!-- tabsets -->
<script>
$(document).ready(function () {
  window.buildTabsets("TOC");
});
</script>

<!-- code folding -->







<div class="fluid-row" id="header">



<h1 class="title toc-ignore">post01-Wei-Li</h1>

</div>


<pre><code>                          Trees in Statistics and R language
  Tree is a widely used data structure in computer science. We have binary search tree, heap, B+ tree, trie and a lot of mutati
  When I looked into the manual of R language, which is one of the best programming language for statistical computing and also
  We will first load the package with library call.
  </code></pre>
<pre class="r"><code>library(rpart)</code></pre>
<p>Then we can grow a classification tree with rpart method and the built-in data frame kyphosis</p>
<pre class="r"><code>fit &lt;- rpart(Kyphosis ~ Age + Number + Start,
    method=&quot;class&quot;, data=kyphosis)</code></pre>
<p>With printcp, we can display the result.</p>
<pre class="r"><code>printcp(fit)</code></pre>
<pre><code>##
## Classification tree:
## rpart(formula = Kyphosis ~ Age + Number + Start, data = kyphosis,
##     method = &quot;class&quot;)
##
## Variables actually used in tree construction:
## [1] Age   Start
```

```
##
## Root node error: 17/81 = 0.20988
##
## n= 81
##
##         CP nsplit rel error  xerror    xstd
## 1 0.176471      0   1.00000 1.00000 0.21559
## 2 0.019608      1   0.82353 0.82353 0.20018
## 3 0.010000      4   0.76471 0.82353 0.20018</code></pre>
<p>Create simple plot with plotcp</p>
<pre class="r"><code>pdf(file = &quot;../images/cp_plot.pdf&quot;)
plotcp(fit)
dev.off()</code></pre>
<pre><code>## quartz_off_screen
##                 2</code></pre>
<p>Summarize the data with summary</p>
<pre class="r"><code>sink(&quot;../output/kyphosis-summary&quot;)
summary(fit)</code></pre>
<pre><code>## Call:
## rpart(formula = Kyphosis ~ Age + Number + Start, data = kyphosis,
##     method = &quot;class&quot;)
##   n= 81
##
##           CP nsplit rel error    xerror      xstd
## 1 0.17647059      0 1.0000000 1.0000000 0.2155872
## 2 0.01960784      1 0.8235294 0.8235294 0.2001751
## 3 0.01000000      4 0.7647059 0.8235294 0.2001751
##
## Variable importance
##  Start    Age Number
##     64     24     12
##
## Node number 1: 81 observations,    complexity param=0.1764706
##   predicted class=absent   expected loss=0.2098765  P(node) =1
##     class counts:    64    17
##    probabilities: 0.790 0.210
##   left son=2 (62 obs) right son=3 (19 obs)
##   Primary splits:
##       Start  &lt; 8.5  to the right, improve=6.762330, (0 missing)
##       Number &lt; 5.5  to the left,  improve=2.866795, (0 missing)
##       Age    &lt; 39.5 to the left,  improve=2.250212, (0 missing)
##   Surrogate splits:
##       Number &lt; 6.5  to the left,  agree=0.802, adj=0.158, (0 split)
##
## Node number 2: 62 observations,    complexity param=0.01960784
##   predicted class=absent   expected loss=0.09677419  P(node) =0.7654321
##     class counts:    56     6
##    probabilities: 0.903 0.097
##   left son=4 (29 obs) right son=5 (33 obs)
##   Primary splits:
##       Start  &lt; 14.5 to the right, improve=1.0205280, (0 missing)
##       Age    &lt; 55   to the left,  improve=0.6848635, (0 missing)
##       Number &lt; 4.5  to the left,  improve=0.2975332, (0 missing)
##   Surrogate splits:
##       Number &lt; 3.5  to the left,  agree=0.645, adj=0.241, (0 split)
##       Age    &lt; 16   to the left,  agree=0.597, adj=0.138, (0 split)
##
## Node number 3: 19 observations
##   predicted class=present  expected loss=0.4210526  P(node) =0.2345679
##     class counts:     8    11
##    probabilities: 0.421 0.579
##
## Node number 4: 29 observations
##   predicted class=absent   expected loss=0  P(node) =0.3580247
##     class counts:    29     0
##    probabilities: 1.000 0.000
##
## Node number 5: 33 observations,    complexity param=0.01960784
##   predicted class=absent   expected loss=0.1818182  P(node) =0.4074074
##     class counts:    27     6
##    probabilities: 0.818 0.182
##   left son=10 (12 obs) right son=11 (21 obs)
##   Primary splits:
##       Age    &lt; 55   to the left,  improve=1.2467530, (0 missing)
##       Start  &lt; 12.5 to the right, improve=0.2887701, (0 missing)
##       Number &lt; 3.5  to the right, improve=0.1753247, (0 missing)
##   Surrogate splits:
##       Start  &lt; 9.5  to the left,  agree=0.758, adj=0.333, (0 split)
##       Number &lt; 5.5  to the right, agree=0.697, adj=0.167, (0 split)
```

```
##       Number &lt; 3.5  to the right, agree=0.697, adj=0.167, (0 split)
##
## Node number 10: 12 observations
##   predicted class=absent   expected loss=0  P(node) =0.1481481
##     class counts:    12     0
##    probabilities: 1.000 0.000
##
## Node number 11: 21 observations,    complexity param=0.01960784
##   predicted class=absent   expected loss=0.2857143  P(node) =0.2592593
##     class counts:    15     6
##    probabilities: 0.714 0.286
##   left son=22 (14 obs) right son=23 (7 obs)
##   Primary splits:
##       Age    &lt; 111  to the right, improve=1.71428600, (0 missing)
##       Start  &lt; 12.5 to the right, improve=0.79365080, (0 missing)
##       Number &lt; 3.5  to the right, improve=0.07142857, (0 missing)
##
## Node number 22: 14 observations
##   predicted class=absent   expected loss=0.1428571  P(node) =0.1728395
##     class counts:    12     2
##    probabilities: 0.857 0.143
##
## Node number 23: 7 observations
##   predicted class=present  expected loss=0.4285714  P(node) =0.08641975
##     class counts:     3     4
##    probabilities: 0.429 0.571</code></pre>
<pre class="r"><code>sink()</code></pre>
<p>Create tree-structured plot that better visualises hierarchical data</p>
<pre class="r"><code>pdf(file = &quot;../images/tree_plot.dpf&quot;)
plot(fit, uniform=TRUE,
    main=&quot;Classification Tree for Kyphosis&quot;)
text(fit, use.n=TRUE, all=TRUE, cex=.8)
dev.off()</code></pre>
<pre><code>## quartz_off_screen
##                 2</code></pre>
<p>To avoid overfitting the data, we can prune the tree to minimize the cross-validated error.</p>
<pre class="r"><code>pfit&lt;- prune(fit, cp=   fit$cptable[which.min(fit$cptable[,&quot;xerror&quot;]),&quot;CP&quot;])</code>
<pre class="r"><code>printcp(pfit)</code></pre>
<pre><code>##
## Classification tree:
## rpart(formula = Kyphosis ~ Age + Number + Start, data = kyphosis,
##     method = &quot;class&quot;)
##
## Variables actually used in tree construction:
## [1] Start
##
## Root node error: 17/81 = 0.20988
##
## n= 81
##
##         CP nsplit rel error  xerror    xstd
## 1 0.176471      0   1.00000 1.00000 0.21559
## 2 0.019608      1   0.82353 0.82353 0.20018</code></pre>
<pre class="r"><code>plotcp(pfit)</code></pre>

<p>Reference: <a href="https://www.statmethods.net/advstats/cart.html" class="uri">https://www.statmethods.net/advstats/cart.ht

</div>

<script>

// add bootstrap table styles to pandoc tables
function bootstrapStylePandocTables() {
  $('tr.header').parent('thead').parent('table').addClass('table table-condensed');
}
$(document).ready(function () {
  bootstrapStylePandocTables();
});


</script>

<!-- dynamically load mathjax for compatibility with self-contained -->
<script>
  (function () {
    var script = document.createElement("script");
```

```
    script.type = "text/javascript";
    script.src  = "https://mathjax.rstudio.com/latest/MathJax.js?config=TeX-AMS-MML_HTMLorMML";
    document.getElementsByTagName("head")[0].appendChild(script);
  })();
</script>

</body>
</html>
```