Item 123

git_comments:

git_commits:

1. **summary:** [SPARK-1041] remove dead code in start script, remind user to set that in spark-env.sh
   message: [SPARK-1041] remove dead code in start script, remind user to set that in spark-env.sh the lines in start-master.sh and start-slave.sh no longer work in ec2, the host name has changed, e.g. ubuntu@ip-172-31-36-93:~$ hostname ip-172-31-36-93 also, the URL to fetch public DNS name also changed, e.g. ubuntu@ip-172-31-36-93:~$ wget -q -O - http://instance-data.ec2.internal/latest/meta-data/public-hostname ubuntu@ip-172-31-36-93:~$ (returns nothing) since we have spark-ec2 project, we don't need to have such ec2-specific lines here, instead, user only need to set in spark-env.sh Author: CodingCat <zhunansjtu@gmail.com> Closes #588 from CodingCat/deadcode_in_sbin and squashes the following commits: e4236e0 [CodingCat] remove dead code in start script, remind user set that in spark-env.sh

github_issues:

github_issues_comments:

github_pulls:

1. **title:** Update GradientDescentSuite.scala
   body: use more faster way to construct an array

github_pulls_comments:

1. Can one of the admins verify this patch?
2. IMHO this will not compile if used like this `1 +: features`, Integers with DenseVector[Double]. Are you sure we this is very useful ? @mengxr
3. Should use `Vectors.dense(1.0 +: features)`, which may be faster than `Vectors.dense(1.0, features: _*)`, depending on how scala compiles the code.
4. @mengxr yupp that would be better.
5. I had modify the code to Vectors.dense(1.0 +: features) . can some setting can make this code compile successful? @mengxr
6. @baishuo change it to `Vectors.dense(1.0 +: features.toArray)`
7. had modified, thanks to @techaddict @mengxr
8. Jenkins, test this please.
9. Merged build triggered.
10. Merged build started.
11. Merged build finished.
12. Refer to this link for build results: https://amplab.cs.berkeley.edu/jenkins/job/SparkPullRequestBuilder/14581/
13. Hi guys @mengxr @techaddict , Is there something need to do before merge? :)
14. (I'd be curious to know why this might or might not be faster, if someone has the time to sketch an explanation. I don't doubt it might be.)
15. @srowen They are the same because in `Vectors.dense(first: Double, others: Double*)` defined, we use `first +: others.toArray` and `toArray` does not copy the data. But it is safer to create an Array before calling `Vectors.dense`. For example, `Array(1.0, values: _*)` is slower than `1.0 +: values` if `values` is an `Array`. See https://github.com/scala/scala/blob/master/src/library/scala/Array.scala#L199
16. LGTM.
17. Merged, thanks.

github_pulls_reviews:

1. This won't compile either, we need to write `features +: 1.0`
2. Same for other changes too.

jira_issues:

jira_issues_comments: