

PKA-MOVIE-RECOMMENDER

December 12, 2023

```
[1]: #import findspark
      #findspark.init()
      import os
      import sys
      import pyspark as ps
      import warnings
      from pyspark.sql import SQLContext
      os.environ['PYSPARK_PYTHON'] = sys.executable
      os.environ['PYSPARK_DRIVER_PYTHON'] = sys.executable
      try:
          # create SparkContext on all CPUs available: in my case I have 4 CPUs on my
          ↪laptop
          sc = ps.SparkContext('local[*]')
          #sqlContext = SQLContext(sc)
          print("Just created a SparkContext")
      except ValueError:
          warnings.warn("SparkContext already exists in this scope")
```

Just created a SparkContext

```
[2]: import unittest
      import sys

      class TestRdd(unittest.TestCase):
          def test_take(self):
              input = sc.parallelize([1,2,3,4])
              self.assertEqual([1,2,3,4], input.take(4))

      def run_tests():
          suite = unittest.TestLoader().loadTestsFromTestCase( TestRdd )
          unittest.TextTestRunner(verbosity=1,stream=sys.stderr).run( suite )

      run_tests()
```

Ran 1 test in 8.709s

OK

```
[3]: help(sc)
```

Help on SparkContext in module pyspark.context object:

```
class SparkContext(builtins.object)
|   SparkContext(master: Optional[str] = None, appName: Optional[str] = None,
sparkHome: Optional[str] = None, pyFiles: Optional[List[str]] = None,
environment: Optional[Dict[str, Any]] = None, batchSize: int = 0, serializer:
'Serializer' = CloudPickleSerializer(), conf: Optional[pyspark.conf.SparkConf] =
None, gateway: Optional[py4j.java_gateway.JavaGateway] = None, jsc:
Optional[py4j.java_gateway.JavaObject] = None, profiler_cls:
Type[pyspark.profiler.BasicProfiler] = <class 'pyspark.profiler.BasicProfiler'>,
udf_profiler_cls: Type[pyspark.profiler.UDFBasicProfiler] = <class
'pyspark.profiler.UDFBasicProfiler'>, memory_profiler_cls:
Type[pyspark.profiler.MemoryProfiler] = <class
'pyspark.profiler.MemoryProfiler'>)
|
|   Main entry point for Spark functionality. A SparkContext represents the
|   connection to a Spark cluster, and can be used to create :class:`RDD` and
|   broadcast variables on that cluster.
|
|   When you create a new SparkContext, at least the master and app name should
|   be set, either through the named parameters here or through `conf`.
|
|   Parameters
|   -----
|   master : str, optional
|       Cluster URL to connect to (e.g. mesos://host:port, spark://host:port,
local[4]).
|   appName : str, optional
|       A name for your job, to display on the cluster web UI.
|   sparkHome : str, optional
|       Location where Spark is installed on cluster nodes.
|   pyFiles : list, optional
|       Collection of .zip or .py files to send to the cluster
|       and add to PYTHONPATH. These can be paths on the local file
|       system or HDFS, HTTP, HTTPS, or FTP URLs.
|   environment : dict, optional
|       A dictionary of environment variables to set on
|       worker nodes.
|   batchSize : int, optional, default 0
|       The number of Python objects represented as a single
|       Java object. Set 1 to disable batching, 0 to automatically choose
|       the batch size based on object sizes, or -1 to use an unlimited
|       batch size
```



```

sparkHome: Optional[str] = None, pyFiles: Optional[List[str]] = None,
environment: Optional[Dict[str, Any]] = None, batchSize: int = 0, serializer:
'Serializer' = CloudPickleSerializer(), conf: Optional[pyspark.conf.SparkConf] =
None, gateway: Optional[py4j.java_gateway.JavaGateway] = None, jsc:
Optional[py4j.java_gateway.JavaObject] = None, profiler_cls:
Type[pyspark.profiler.BasicProfiler] = <class 'pyspark.profiler.BasicProfiler'>,
udf_profiler_cls: Type[pyspark.profiler.UDFBasicProfiler] = <class
'pyspark.profiler.UDFBasicProfiler'>, memory_profiler_cls:
Type[pyspark.profiler.MemoryProfiler] = <class
'pyspark.profiler.MemoryProfiler'>)
|     Initialize self. See help(type(self)) for accurate signature.
|
|     __repr__(self) -> str
|         Return repr(self).
|
|     accumulator(self, value: ~T, accum_param:
Optional[ForwardRef('AccumulatorParam[T]')] = None) -> 'Accumulator[T]'
|         Create an :class:`Accumulator` with the given initial value, using a
given
|         :class:`AccumulatorParam` helper object to define how to add values of
the
|         data type if provided. Default AccumulatorParams are used for integers
|         and floating-point numbers if you do not provide one. For other types,
|         a custom AccumulatorParam can be used.
|
|         .. versionadded:: 0.7.0
|
|         Parameters
|         -----
|         value : T
|             initialized value
|         accum_param : :class:`pyspark.AccumulatorParam`, optional
|             helper object to define how to add values
|
|         Returns
|         -----
|         :class:`Accumulator`
|             `Accumulator` object, a shared variable that can be accumulated
|
|         Examples
|         -----
|         >>> acc = sc.accumulator(9)
|         >>> acc.value
|         9
|         >>> acc += 1
|         >>> acc.value
|         10
|

```

```

|     Accumulator object can be accumulated in RDD operations:
|
|     >>> rdd = sc.range(5)
|     >>> def f(x):
|     ...     global acc
|     ...     acc += 1
|     >>> rdd.foreach(f)
|     >>> acc.value
|     15
|
|     addArchive(self, path: str) -> None
|         Add an archive to be downloaded with this Spark job on every node.
|         The `path` passed can be either a local file, a file in HDFS
|         (or other Hadoop-supported filesystems), or an HTTP, HTTPS or
|         FTP URI.
|
|         To access the file in Spark jobs, use :meth:`SparkFiles.get` with the
|         filename to find its download/unpacked location. The given path should
|         be one of .zip, .tar, .tar.gz, .tgz and .jar.
|
|         .. versionadded:: 3.3.0
|
|         Parameters
|         -----
|         path : str
|             can be either a local file, a file in HDFS (or other Hadoop-
supported
|             filesystems), or an HTTP, HTTPS or FTP URI. To access the file in
Spark jobs,
|             use :meth:`SparkFiles.get` to find its download location.
|
|         See Also
|         -----
|         :meth:`SparkContext.listArchives`
|         :meth:`SparkFiles.get`
|
|         Notes
|         -----
|         A path can be added only once. Subsequent additions of the same path are
ignored.
|         This API is experimental.
|
|         Examples
|         -----
|         Creates a zipped file that contains a text file written '100'.
|
|         >>> import os
|         >>> import tempfile

```

```

|     >>> import zipfile
|     >>> from pyspark import SparkFiles
|
|     >>> with tempfile.TemporaryDirectory() as d:
|     ...     path = os.path.join(d, "test.txt")
|     ...     with open(path, "w") as f:
|     ...         _ = f.write("100")
|     ...
|     ...     zip_path1 = os.path.join(d, "test1.zip")
|     ...     with zipfile.ZipFile(zip_path1, "w", zipfile.ZIP_DEFLATED) as z:
|     ...         z.write(path, os.path.basename(path))
|     ...
|     ...     zip_path2 = os.path.join(d, "test2.zip")
|     ...     with zipfile.ZipFile(zip_path2, "w", zipfile.ZIP_DEFLATED) as z:
|     ...         z.write(path, os.path.basename(path))
|     ...
|     ...     sc.addArchive(zip_path1)
|     ...     arch_list1 = sorted(sc.listArchives)
|     ...
|     ...     sc.addArchive(zip_path2)
|     ...     arch_list2 = sorted(sc.listArchives)
|     ...
|     ...     # add zip_path2 twice, this addition will be ignored
|     ...     sc.addArchive(zip_path2)
|     ...     arch_list3 = sorted(sc.listArchives)
|     ...
|     ...     def func(iterator):
|     ...         with open("%s/test.txt" % SparkFiles.get("test1.zip")) as f:
|     ...             mul = int(f.readline())
|     ...             return [x * mul for x in iterator]
|     ...
|     ...     collected = sc.parallelize([1, 2, 3,
4]) .mapPartitions(func).collect()

```

```

|     >>> arch_list1
|     ['file:/.../test1.zip']
|     >>> arch_list2
|     ['file:/.../test1.zip', 'file:/.../test2.zip']
|     >>> arch_list3
|     ['file:/.../test1.zip', 'file:/.../test2.zip']
|     >>> collected
|     [100, 200, 300, 400]

```

```

|     addFile(self, path: str, recursive: bool = False) -> None
|         Add a file to be downloaded with this Spark job on every node.
|         The `path` passed can be either a local file, a file in HDFS
|         (or other Hadoop-supported filesystems), or an HTTP, HTTPS or
|         FTP URI.

```

```

|
|     To access the file in Spark jobs, use :meth:`SparkFiles.get` with the
|     filename to find its download location.
|
|     A directory can be given if the recursive option is set to True.
|     Currently directories are only supported for Hadoop-supported
fileSystems.
|
|     .. versionadded:: 0.7.0
|
|     Parameters
|     -----
|     path : str
|         can be either a local file, a file in HDFS (or other Hadoop-
supported
|         fileSystems), or an HTTP, HTTPS or FTP URI. To access the file in
Spark jobs,
|         use :meth:`SparkFiles.get` to find its download location.
|     recursive : bool, default False
|         whether to recursively add files in the input directory
|
|     See Also
|     -----
|     :meth:`SparkContext.listFiles`
|     :meth:`SparkContext.addPyFile`
|     :meth:`SparkFiles.get`
|
|     Notes
|     ----
|     A path can be added only once. Subsequent additions of the same path are
ignored.
|
|     Examples
|     -----
|     >>> import os
|     >>> import tempfile
|     >>> from pyspark import SparkFiles
|
|     >>> with tempfile.TemporaryDirectory() as d:
|     ...     path1 = os.path.join(d, "test1.txt")
|     ...     with open(path1, "w") as f:
|     ...         _ = f.write("100")
|     ...
|     ...     path2 = os.path.join(d, "test2.txt")
|     ...     with open(path2, "w") as f:
|     ...         _ = f.write("200")
|     ...
|     ...     sc.addFile(path1)

```

```

|         ...     file_list1 = sorted(sc.listFiles)
|         ...
|         ...     sc.addFile(path2)
|         ...     file_list2 = sorted(sc.listFiles)
|         ...
|         ...     # add path2 twice, this addition will be ignored
|         ...     sc.addFile(path2)
|         ...     file_list3 = sorted(sc.listFiles)
|         ...
|         ...     def func(iterator):
|         ...         with open(SparkFiles.get("test1.txt")) as f:
|         ...             mul = int(f.readline())
|         ...             return [x * mul for x in iterator]
|         ...
|         ...     collected = sc.parallelize([1, 2, 3,
4]).mapPartitions(func).collect()
|
|         >>> file_list1
|         ['file:/.../test1.txt']
|         >>> file_list2
|         ['file:/.../test1.txt', 'file:/.../test2.txt']
|         >>> file_list3
|         ['file:/.../test1.txt', 'file:/.../test2.txt']
|         >>> collected
|         [100, 200, 300, 400]
|
|         addPyFile(self, path: str) -> None
|         Add a .py or .zip dependency for all tasks to be executed on this
|         SparkContext in the future. The `path` passed can be either a local
|         file, a file in HDFS (or other Hadoop-supported filesystems), or an
|         HTTP, HTTPS or FTP URI.
|
|         .. versionadded:: 0.7.0
|
|         Parameters
|         -----
|         path : str
|             can be either a .py file or .zip dependency.
|
|         See Also
|         -----
|         :meth:`SparkContext.addFile`
|
|         Notes
|         -----
|         A path can be added only once. Subsequent additions of the same path are
|         ignored.
|

```



```

|   binaryFiles(self, path: str, minPartitions: Optional[int] = None) ->
pyspark.rdd.RDD[typing.Tuple[str, bytes]]
|       Read a directory of binary files from HDFS, a local file system
|       (available on all nodes), or any Hadoop-supported file system URI
|       as a byte array. Each file is read as a single record and returned
|       in a key-value pair, where the key is the path of each file, the
|       value is the content of each file.
|
|       .. versionadded:: 1.3.0
|
|       Parameters
|       -----
|       path : str
|           directory to the input data files, the path can be comma separated
|           paths as a list of inputs
|       minPartitions : int, optional
|           suggested minimum number of partitions for the resulting RDD
|
|       Returns
|       -----
|       :class:`RDD`
|           RDD representing path-content pairs from the file(s).
|
|       Notes
|       ----
|       Small files are preferred, large file is also allowable, but may cause
bad performance.
|
|       See Also
|       -----
|       :meth:`SparkContext.binaryRecords`
|
|       Examples
|       -----
|       >>> import os
|       >>> import tempfile
|       >>> with tempfile.TemporaryDirectory() as d:
|       ...     # Write a temporary binary file
|       ...     with open(os.path.join(d, "1.bin"), "wb") as f1:
|       ...         _ = f1.write(b"binary data I")
|       ...
|       ...     # Write another temporary binary file
|       ...     with open(os.path.join(d, "2.bin"), "wb") as f2:
|       ...         _ = f2.write(b"binary data II")
|       ...
|       ...     collected = sorted(sc.binaryFiles(d).collect())
|
|       >>> collected

```

```

|         [('.../1.bin', b'binary data I'), ('.../2.bin', b'binary data II')]
|
|     binaryRecords(self, path: str, recordLength: int) -> pyspark.rdd.RDD[bytes]
|         Load data from a flat binary file, assuming each record is a set of
numbers
|         with the specified numerical format (see ByteBuffer), and the number of
|         bytes per record is constant.
|
|         .. versionadded:: 1.3.0
|
|     Parameters
|     -----
|     path : str
|         Directory to the input data files
|     recordLength : int
|         The length at which to split the records
|
|     Returns
|     -----
|     :class:`RDD`
|         RDD of data with values, represented as byte arrays
|
|     See Also
|     -----
|     :meth:`SparkContext.binaryFiles`
|
|     Examples
|     -----
|     >>> import os
|     >>> import tempfile
|     >>> with tempfile.TemporaryDirectory() as d:
|     ...     # Write a temporary file
|     ...     with open(os.path.join(d, "1.bin"), "w") as f:
|     ...         for i in range(3):
|     ...             _ = f.write("%04d" % i)
|     ...
|     ...     # Write another file
|     ...     with open(os.path.join(d, "2.bin"), "w") as f:
|     ...         for i in [-1, -2, -10]:
|     ...             _ = f.write("%04d" % i)
|     ...
|     ...     collected = sorted(sc.binaryRecords(d, 4).collect())
|
|     >>> collected
|     [b'-001', b'-002', b'-010', b'0000', b'0001', b'0002']
|
|     broadcast(self, value: ~T) -> 'Broadcast[T]'
|         Broadcast a read-only variable to the cluster, returning a

```

```

:class:`Broadcast`
|     object for reading it in distributed functions. The variable will
|     be sent to each cluster only once.
|
|     .. versionadded:: 0.7.0
|
|     Parameters
|     -----
|     value : T
|         value to broadcast to the Spark nodes
|
|     Returns
|     -----
|     :class:`Broadcast`
|         :class:`Broadcast` object, a read-only variable cached on each
machine
|
|     Examples
|     -----
|     >>> mapping = {1: 10001, 2: 10002}
|     >>> bc = sc.broadcast(mapping)
|
|     >>> rdd = sc.range(5)
|     >>> rdd2 = rdd.map(lambda i: bc.value[i] if i in bc.value else -1)
|     >>> rdd2.collect()
|     [-1, 10001, 10002, -1, -1]
|
|     >>> bc.destroy()
|
|     cancelAllJobs(self) -> None
|         Cancel all jobs that have been scheduled or are running.
|
|     .. versionadded:: 1.1.0
|
|     See Also
|     -----
|     :meth:`SparkContext.cancelJobGroup`
|     :meth:`SparkContext.runJob`
|
|     cancelJobGroup(self, groupId: str) -> None
|         Cancel active jobs for the specified group. See
:meth:`SparkContext.setJobGroup`.
|         for more information.
|
|     .. versionadded:: 1.1.0
|
|     Parameters
|     -----

```

```

|     groupId : str
|         The group ID to cancel the job.
|
|     See Also
|     -----
|     :meth:`SparkContext.setJobGroup`
|     :meth:`SparkContext.cancelJobGroup`
|
| dump_profiles(self, path: str) -> None
|     Dump the profile stats into directory `path`
|
|     .. versionadded:: 1.2.0
|
|     See Also
|     -----
|     :meth:`SparkContext.show_profiles`
|
| emptyRDD(self) -> pyspark.rdd.RDD[typing.Any]
|     Create an :class:`RDD` that has no partitions or elements.
|
|     .. versionadded:: 1.5.0
|
|     Returns
|     -----
|     :class:`RDD`
|         An empty RDD
|
|     Examples
|     -----
|     >>> sc.emptyRDD()
|     EmptyRDD...
|     >>> sc.emptyRDD().count()
|     0
|
| getCheckpointDir(self) -> Optional[str]
|     Return the directory where RDDs are checkpointed. Returns None if no
|     checkpoint directory has been set.
|
|     .. versionadded:: 3.1.0
|
|     See Also
|     -----
|     :meth:`SparkContext.setCheckpointDir`
|     :meth:`RDD.checkpoint`
|     :meth:`RDD.getCheckpointFile`
|
| getConf(self) -> pyspark.conf.SparkConf
|     Return a copy of this SparkContext's configuration :class:`SparkConf`.

```

```

|
| .. versionadded:: 2.1.0
|
| getLocalProperty(self, key: str) -> Optional[str]
|     Get a local property set in this thread, or null if it is missing. See
|     :meth:`setLocalProperty`.
|
| .. versionadded:: 1.0.0
|
| See Also
| -----
| :meth:`SparkContext.setLocalProperty`
|
| hadoopFile(self, path: str, inputFormatClass: str, keyClass: str,
valueClass: str, keyConverter: Optional[str] = None, valueConverter:
Optional[str] = None, conf: Optional[Dict[str, str]] = None, batchSize: int = 0)
-> pyspark.rdd.RDD[typing.Tuple[~T, ~U]]
|     Read an 'old' Hadoop InputFormat with arbitrary key and value class from
HDFS,
|     a local file system (available on all nodes), or any Hadoop-supported
file system URI.
|     The mechanism is the same as for :meth:`SparkContext.sequenceFile`.
|
| .. versionadded:: 1.1.0
|
|     A Hadoop configuration can be passed in as a Python dict. This will be
converted into a
|     Configuration in Java.
|
| Parameters
| -----
| path : str
|     path to Hadoop file
| inputFormatClass : str
|     fully qualified classname of Hadoop InputFormat
|     (e.g. "org.apache.hadoop.mapreduce.lib.input.TextInputFormat")
| keyClass : str
|     fully qualified classname of key Writable class (e.g.
"org.apache.hadoop.io.Text")
| valueClass : str
|     fully qualified classname of value Writable class
|     (e.g. "org.apache.hadoop.io.LongWritable")
| keyConverter : str, optional
|     fully qualified name of a function returning key WritableConverter
| valueConverter : str, optional
|     fully qualified name of a function returning value WritableConverter
| conf : dict, optional
|     Hadoop configuration, passed in as a dict

```

```

|     batchSize : int, optional, default 0
|         The number of Python objects represented as a single
|         Java object. (default 0, choose batchSize automatically)
|
| Returns
| -----
| :class:`RDD`
|     RDD of tuples of key and corresponding value
|
| See Also
| -----
| :meth:`RDD.saveAsSequenceFile`
| :meth:`RDD.saveAsNewAPIHadoopFile`
| :meth:`RDD.saveAsHadoopFile`
| :meth:`SparkContext.newAPIHadoopFile`
| :meth:`SparkContext.hadoopRDD`
|
| Examples
| -----
| >>> import os
| >>> import tempfile
|
| Set the related classes
|
| >>> output_format_class = "org.apache.hadoop.mapred.TextOutputFormat"
| >>> input_format_class = "org.apache.hadoop.mapred.TextInputFormat"
| >>> key_class = "org.apache.hadoop.io.IntWritable"
| >>> value_class = "org.apache.hadoop.io.Text"
|
| >>> with tempfile.TemporaryDirectory() as d:
| ...     path = os.path.join(d, "old_hadoop_file")
| ...
| ...     # Write a temporary Hadoop file
| ...     rdd = sc.parallelize([(1, ""), (1, "a"), (3, "x")])
| ...     rdd.saveAsHadoopFile(path, output_format_class, key_class,
value_class)
| ...
| ...     loaded = sc.hadoopFile(path, input_format_class, key_class,
value_class)
| ...     collected = sorted(loaded.collect())
|
| >>> collected
| [(0, '1\t'), (0, '1\ta'), (0, '3\tx')]
|
|     hadoopRDD(self, inputFormatClass: str, keyClass: str, valueClass: str,
keyConverter: Optional[str] = None, valueConverter: Optional[str] = None, conf:
Optional[Dict[str, str]] = None, batchSize: int = 0) ->
pyspark.rdd.RDD[typing.Tuple[~T, ~U]]

```

```

|         Read an 'old' Hadoop InputFormat with arbitrary key and value class,
from an arbitrary
|         Hadoop configuration, which is passed in as a Python dict.
|         This will be converted into a Configuration in Java.
|         The mechanism is the same as for meth:`SparkContext.sequenceFile`.
|
|         .. versionadded:: 1.1.0
|
|         Parameters
|         -----
|         inputFormatClass : str
|             fully qualified classname of Hadoop InputFormat
|             (e.g. "org.apache.hadoop.mapreduce.lib.input.TextInputFormat")
|         keyClass : str
|             fully qualified classname of key Writable class (e.g.
"org.apache.hadoop.io.Text")
|         valueClass : str
|             fully qualified classname of value Writable class
|             (e.g. "org.apache.hadoop.io.LongWritable")
|         keyConverter : str, optional
|             fully qualified name of a function returning key WritableConverter
|         valueConverter : str, optional
|             fully qualified name of a function returning value WritableConverter
|         conf : dict, optional
|             Hadoop configuration, passed in as a dict
|         batchSize : int, optional, default 0
|             The number of Python objects represented as a single
|             Java object. (default 0, choose batchSize automatically)
|
|         Returns
|         -----
|         :class:`RDD`
|             RDD of tuples of key and corresponding value
|
|         See Also
|         -----
|         :meth:`RDD.saveAsNewAPIHadoopDataset`
|         :meth:`RDD.saveAsHadoopDataset`
|         :meth:`SparkContext.newAPIHadoopRDD`
|         :meth:`SparkContext.hadoopFile`
|
|         Examples
|         -----
|         >>> import os
|         >>> import tempfile
|
|         Set the related classes

```

```

|     >>> output_format_class = "org.apache.hadoop.mapred.TextOutputFormat"
|     >>> input_format_class = "org.apache.hadoop.mapred.TextInputFormat"
|     >>> key_class = "org.apache.hadoop.io.IntWritable"
|     >>> value_class = "org.apache.hadoop.io.Text"
|
|     >>> with tempfile.TemporaryDirectory() as d:
|     ...     path = os.path.join(d, "old_hadoop_file")
|     ...
|     ...     # Create the conf for writing
|     ...     write_conf = {
|     ...         "mapred.output.format.class": output_format_class,
|     ...         "mapreduce.job.output.key.class": key_class,
|     ...         "mapreduce.job.output.value.class": value_class,
|     ...         "mapreduce.output.fileoutputformat.outputdir": path,
|     ...     }
|     ...
|     ...     # Write a temporary Hadoop file
|     ...     rdd = sc.parallelize([(1, ""), (1, "a"), (3, "x")])
|     ...     rdd.saveAsHadoopDataset(conf=write_conf)
|     ...
|     ...     # Create the conf for reading
|     ...     read_conf = {"mapreduce.input.fileinputformat.inputdir": path}
|     ...
|     ...     loaded = sc.hadoopRDD(input_format_class, key_class,
value_class, conf=read_conf)
|     ...     collected = sorted(loaded.collect())
|
|     >>> collected
|     [(0, '1\t'), (0, '1\ta'), (0, '3\tx')]
|
|     newAPIHadoopFile(self, path: str, inputFormatClass: str, keyClass: str,
valueClass: str, keyConverter: Optional[str] = None, valueConverter:
Optional[str] = None, conf: Optional[Dict[str, str]] = None, batchSize: int = 0)
-> pyspark.rdd.RDD[typing.Tuple[~T, ~U]]
|
|     Read a 'new API' Hadoop InputFormat with arbitrary key and value class
from HDFS,
|
|     a local file system (available on all nodes), or any Hadoop-supported
file system URI.
|
|     The mechanism is the same as for meth:`SparkContext.sequenceFile`.
|
|     A Hadoop configuration can be passed in as a Python dict. This will be
converted into a
|
|     Configuration in Java
|
|     .. versionadded:: 1.1.0
|
|     Parameters
|     -----

```



```

|     path : str
|         path to Hadoop file
|     inputFormatClass : str
|         fully qualified classname of Hadoop InputFormat
|         (e.g. "org.apache.hadoop.mapreduce.lib.input.TextInputFormat")
|     keyClass : str
|         fully qualified classname of key Writable class
|         (e.g. "org.apache.hadoop.io.Text")
|     valueClass : str
|         fully qualified classname of value Writable class
|         (e.g. "org.apache.hadoop.io.LongWritable")
|     keyConverter : str, optional
|         fully qualified name of a function returning key WritableConverter
|         None by default
|     valueConverter : str, optional
|         fully qualified name of a function returning value WritableConverter
|         None by default
|     conf : dict, optional
|         Hadoop configuration, passed in as a dict
|         None by default
|     batchSize : int, optional, default 0
|         The number of Python objects represented as a single
|         Java object. (default 0, choose batchSize automatically)
|
| Returns
| -----
| :class:`RDD`
|     RDD of tuples of key and corresponding value
|
| See Also
| -----
| :meth:`RDD.saveAsSequenceFile`
| :meth:`RDD.saveAsNewAPIHadoopFile`
| :meth:`RDD.saveAsHadoopFile`
| :meth:`SparkContext.sequenceFile`
| :meth:`SparkContext.hadoopFile`
|
| Examples
| -----
| >>> import os
| >>> import tempfile
|
| Set the related classes
|
| >>> output_format_class =
"org.apache.hadoop.mapreduce.lib.output.SequenceFileOutputFormat"
|     >>> input_format_class =
"org.apache.hadoop.mapreduce.lib.input.SequenceFileInputFormat"

```

```

|     >>> key_class = "org.apache.hadoop.io.IntWritable"
|     >>> value_class = "org.apache.hadoop.io.Text"
|
|     >>> with tempfile.TemporaryDirectory() as d:
|         ...     path = os.path.join(d, "new_hadoop_file")
|         ...
|         ...     # Write a temporary Hadoop file
|         ...     rdd = sc.parallelize([(1, ""), (1, "a"), (3, "x")])
|         ...     rdd.saveAsNewAPIHadoopFile(path, output_format_class, key_class,
value_class)
|         ...
|         ...     loaded = sc.newAPIHadoopFile(path, input_format_class,
key_class, value_class)
|         ...     collected = sorted(loaded.collect())
|
|     >>> collected
|     [(1, ''), (1, 'a'), (3, 'x')]
|
|     newAPIHadoopRDD(self, inputFormatClass: str, keyClass: str, valueClass: str,
keyConverter: Optional[str] = None, valueConverter: Optional[str] = None, conf:
Optional[Dict[str, str]] = None, batchSize: int = 0) ->
pyspark.rdd.RDD[typing.Tuple[~T, ~U]]
|     Read a 'new API' Hadoop InputFormat with arbitrary key and value class,
from an arbitrary
|     Hadoop configuration, which is passed in as a Python dict.
|     This will be converted into a Configuration in Java.
|     The mechanism is the same as for meth:`SparkContext.sequenceFile`.
|
|     .. versionadded:: 1.1.0
|
|     Parameters
|     -----
|     inputFormatClass : str
|         fully qualified classname of Hadoop InputFormat
|         (e.g. "org.apache.hadoop.mapreduce.lib.input.TextInputFormat")
|     keyClass : str
|         fully qualified classname of key Writable class (e.g.
"org.apache.hadoop.io.Text")
|     valueClass : str
|         fully qualified classname of value Writable class
|         (e.g. "org.apache.hadoop.io.LongWritable")
|     keyConverter : str, optional
|         fully qualified name of a function returning key WritableConverter
|         (None by default)
|     valueConverter : str, optional
|         fully qualified name of a function returning value WritableConverter
|         (None by default)
|     conf : dict, optional

```

```

|         Hadoop configuration, passed in as a dict (None by default)
| batchSize : int, optional, default 0
|         The number of Python objects represented as a single
|         Java object. (default 0, choose batchSize automatically)
|
| Returns
| -----
| :class:`RDD`
|     RDD of tuples of key and corresponding value
|
| See Also
| -----
| :meth:`RDD.saveAsNewAPIHadoopDataset`
| :meth:`RDD.saveAsHadoopDataset`
| :meth:`SparkContext.hadoopRDD`
| :meth:`SparkContext.hadoopFile`
|
| Examples
| -----
| >>> import os
| >>> import tempfile
|
| Set the related classes
|
| >>> output_format_class =
"org.apache.hadoop.mapreduce.lib.output.SequenceFileOutputFormat"
| >>> input_format_class =
"org.apache.hadoop.mapreduce.lib.input.SequenceFileInputFormat"
| >>> key_class = "org.apache.hadoop.io.IntWritable"
| >>> value_class = "org.apache.hadoop.io.Text"
|
| >>> with tempfile.TemporaryDirectory() as d:
| ...     path = os.path.join(d, "new_hadoop_file")
| ...
| ...     # Create the conf for writing
| ...     write_conf = {
| ...         "mapreduce.job.outputformat.class": (output_format_class),
| ...         "mapreduce.job.output.key.class": key_class,
| ...         "mapreduce.job.output.value.class": value_class,
| ...         "mapreduce.output.fileoutputformat.outputdir": path,
| ...     }
| ...
| ...     # Write a temporary Hadoop file
| ...     rdd = sc.parallelize([(1, ""), (1, "a"), (3, "x")])
| ...     rdd.saveAsNewAPIHadoopDataset(conf=write_conf)
| ...
| ...     # Create the conf for reading
| ...     read_conf = {"mapreduce.input.fileinputformat.inputdir": path}

```

```

|         ...
|         ...     loaded = sc.newAPIHadoopRDD(input_format_class,
|         ...         key_class, value_class, conf=read_conf)
|         ...     collected = sorted(loaded.collect())
|
|     >>> collected
|     [(1, ''), (1, 'a'), (3, 'x')]
|
|     parallelize(self, c: Iterable[~T], numSlices: Optional[int] = None) ->
pyspark.rdd.RDD[~T]
|         Distribute a local Python collection to form an RDD. Using range
|         is recommended if the input represents a range for performance.
|
|         .. versionadded:: 0.7.0
|
|         Parameters
|         -----
|         c : :class:`collections.abc.Iterable`
|             iterable collection to distribute
|         numSlices : int, optional
|             the number of partitions of the new RDD
|
|         Returns
|         -----
|         :class:`RDD`
|             RDD representing distributed collection.
|
|         Examples
|         -----
|         >>> sc.parallelize([0, 2, 3, 4, 6], 5).glom().collect()
|         [[0], [2], [3], [4], [6]]
|         >>> sc.parallelize(range(0, 6, 2), 5).glom().collect()
|         [[], [0], [], [2], [4]]
|
|         Deal with a list of strings.
|
|         >>> strings = ["a", "b", "c"]
|         >>> sc.parallelize(strings, 2).glom().collect()
|         [['a'], ['b', 'c']]
|
|     pickleFile(self, name: str, minPartitions: Optional[int] = None) ->
pyspark.rdd.RDD[typing.Any]
|         Load an RDD previously saved using :meth:`RDD.saveAsPickleFile` method.
|
|         .. versionadded:: 1.1.0
|
|         Parameters
|         -----

```

```

|     name : str
|         directory to the input data files, the path can be comma separated
|         paths as a list of inputs
|     minPartitions : int, optional
|         suggested minimum number of partitions for the resulting RDD
|
| Returns
| -----
| :class:`RDD`
|     RDD representing unpickled data from the file(s).
|
| See Also
| -----
| :meth:`RDD.saveAsPickleFile`
|
| Examples
| -----
| >>> import os
| >>> import tempfile
| >>> with tempfile.TemporaryDirectory() as d:
| ...     # Write a temporary pickled file
| ...     path1 = os.path.join(d, "pickled1")
| ...     sc.parallelize(range(10)).saveAsPickleFile(path1, 3)
| ...
| ...     # Write another temporary pickled file
| ...     path2 = os.path.join(d, "pickled2")
| ...     sc.parallelize(range(-10, -5)).saveAsPickleFile(path2, 3)
| ...
| ...     # Load picked file
| ...     collected1 = sorted(sc.pickleFile(path1, 3).collect())
| ...     collected2 = sorted(sc.pickleFile(path2, 4).collect())
| ...
| ...     # Load two picked files together
| ...     collected3 = sorted(sc.pickleFile('{}{}'.format(path1, path2),
5).collect())
|
| >>> collected1
| [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
| >>> collected2
| [-10, -9, -8, -7, -6]
| >>> collected3
| [-10, -9, -8, -7, -6, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
|
|     range(self, start: int, end: Optional[int] = None, step: int = 1, numSlices:
Optional[int] = None) -> pyspark.rdd.RDD[int]
|         Create a new RDD of int containing elements from `start` to `end`
|         (exclusive), increased by `step` every element. Can be called the same
|         way as python's built-in range() function. If called with a single

```

```

argument,
|     the argument is interpreted as `end`, and `start` is set to 0.
|
|     .. versionadded:: 1.5.0
|
| Parameters
| -----
| start : int
|         the start value
| end : int, optional
|         the end value (exclusive)
| step : int, optional, default 1
|         the incremental step
| numSlices : int, optional
|         the number of partitions of the new RDD
|
| Returns
| -----
| :class:`RDD`
|     An RDD of int
|
| See Also
| -----
| :meth:`pyspark.sql.Session.range`
|
| Examples
| -----
| >>> sc.range(5).collect()
| [0, 1, 2, 3, 4]
| >>> sc.range(2, 4).collect()
| [2, 3]
| >>> sc.range(1, 7, 2).collect()
| [1, 3, 5]
|
| Generate RDD with a negative step
|
| >>> sc.range(5, 0, -1).collect()
| [5, 4, 3, 2, 1]
| >>> sc.range(0, 5, -1).collect()
| []
|
| Control the number of partitions
|
| >>> sc.range(5, numSlices=1).getNumPartitions()
| 1
| >>> sc.range(5, numSlices=10).getNumPartitions()
| 10

```

```

|   runJob(self, rdd: pyspark.rdd.RDD[~T], partitionFunc:
Callable[[Iterable[~T]], Iterable[~U]], partitions: Optional[Sequence[int]] =
None, allowLocal: bool = False) -> List[~U]
|       Executes the given partitionFunc on the specified set of partitions,
|       returning the result as an array of elements.
|
|       If 'partitions' is not specified, this will run over all partitions.
|
|       .. versionadded:: 1.1.0
|
|       Parameters
|       -----
|       rdd : :class:`RDD`
|           target RDD to run tasks on
|       partitionFunc : function
|           a function to run on each partition of the RDD
|       partitions : list, optional
|           set of partitions to run on; some jobs may not want to compute on
all
|           partitions of the target RDD, e.g. for operations like `first`
|       allowLocal : bool, default False
|           this parameter takes no effect
|
|       Returns
|       -----
|       list
|           results of specified partitions
|
|       See Also
|       -----
|       :meth:`SparkContext.cancelAllJobs`
|
|       Examples
|       -----
|       >>> myRDD = sc.parallelize(range(6), 3)
|       >>> sc.runJob(myRDD, lambda part: [x * x for x in part])
|       [0, 1, 4, 9, 16, 25]
|
|       >>> myRDD = sc.parallelize(range(6), 3)
|       >>> sc.runJob(myRDD, lambda part: [x * x for x in part], [0, 2], True)
|       [0, 1, 16, 25]
|
|   sequenceFile(self, path: str, keyClass: Optional[str] = None, valueClass:
Optional[str] = None, keyConverter: Optional[str] = None, valueConverter:
Optional[str] = None, minSplits: Optional[int] = None, batchSize: int = 0) ->
pyspark.rdd.RDD[typing.Tuple[~T, ~U]]
|       Read a Hadoop SequenceFile with arbitrary key and value Writable class
from HDFS,

```

```

|         a local file system (available on all nodes), or any Hadoop-supported
file system URI.
|         The mechanism is as follows:
|
|         1. A Java RDD is created from the SequenceFile or other InputFormat,
and the key
|             and value Writable classes
|         2. Serialization is attempted via Pickle pickling
|         3. If this fails, the fallback is to call 'toString' on each key and
value
|         4. :class:`CPickleSerializer` is used to deserialize pickled objects
on the Python side
|
|         .. versionadded:: 1.3.0
|
|         Parameters
|         -----
|         path : str
|             path to sequencefile
|         keyClass: str, optional
|             fully qualified classname of key Writable class (e.g.
"org.apache.hadoop.io.Text")
|         valueClass : str, optional
|             fully qualified classname of value Writable class
|             (e.g. "org.apache.hadoop.io.LongWritable")
|         keyConverter : str, optional
|             fully qualified name of a function returning key WritableConverter
|         valueConverter : str, optional
|             fully qualifiedname of a function returning value WritableConverter
|         minSplits : int, optional
|             minimum splits in dataset (default min(2, sc.defaultParallelism))
|         batchSize : int, optional, default 0
|             The number of Python objects represented as a single
|             Java object. (default 0, choose batchSize automatically)
|
|         Returns
|         -----
|         :class:`RDD`
|             RDD of tuples of key and corresponding value
|
|         See Also
|         -----
|         :meth:`RDD.saveAsSequenceFile`
|         :meth:`RDD.saveAsNewAPIHadoopFile`
|         :meth:`RDD.saveAsHadoopFile`
|         :meth:`SparkContext.newAPIHadoopFile`
|         :meth:`SparkContext.hadoopFile`

```



```

|     Examples
|     -----
|     >>> import os
|     >>> import tempfile
|
|     Set the class of output format
|
|     >>> output_format_class =
"org.apache.hadoop.mapreduce.lib.output.SequenceFileOutputFormat"
|
|     >>> with tempfile.TemporaryDirectory() as d:
|     ...     path = os.path.join(d, "hadoop_file")
|     ...
|     ...     # Write a temporary Hadoop file
|     ...     rdd = sc.parallelize([(1, {3.0: "bb"}), (2, {1.0: "aa"}), (3,
{2.0: "dd"})])
|     ...     rdd.saveAsNewAPIHadoopFile(path, output_format_class)
|     ...
|     ...     collected = sorted(sc.sequenceFile(path).collect())
|
|     >>> collected
|     [(1, {3.0: 'bb'}), (2, {1.0: 'aa'}), (3, {2.0: 'dd'})]
|
| setCheckpointDir(self, dirName: str) -> None
|     Set the directory under which RDDs are going to be checkpointed. The
|     directory must be an HDFS path if running on a cluster.
|
|     .. versionadded:: 0.7.0
|
|     Parameters
|     -----
|     dirName : str
|         path to the directory where checkpoint files will be stored
|         (must be HDFS path if running in cluster)
|
|     See Also
|     -----
|     :meth:`SparkContext.getCheckpointDir`
|     :meth:`RDD.checkpoint`
|     :meth:`RDD.getCheckpointFile`
|
| setJobDescription(self, value: str) -> None
|     Set a human readable description of the current job.
|
|     .. versionadded:: 2.3.0
|
|     Parameters
|     -----

```

```

|         value : str
|             The job description to set.
|
|         Notes
|         -----
|         If you run jobs in parallel, use :class:`pyspark.InheritableThread` for
thread
|         local inheritance.
|
|         setJobGroup(self, groupId: str, description: str, interruptOnCancel: bool =
False) -> None
|             Assigns a group ID to all the jobs started by this thread until the
group ID is set to a
|             different value or cleared.
|
|             Often, a unit of execution in an application consists of multiple Spark
actions or jobs.
|             Application programmers can use this method to group all those jobs
together and give a
|             group description. Once set, the Spark web UI will associate such jobs
with this group.
|
|             The application can use :meth:`SparkContext.cancelJobGroup` to cancel
all
|             running jobs in this group.
|
|             .. versionadded:: 1.0.0
|
|         Parameters
|         -----
|         groupId : str
|             The group ID to assign.
|         description : str
|             The description to set for the job group.
|         interruptOnCancel : bool, optional, default False
|             whether to interrupt jobs on job cancellation.
|
|         Notes
|         -----
|         If interruptOnCancel is set to true for the job group, then job
cancellation will result
|         in Thread.interrupt() being called on the job's executor threads. This
is useful to help
|         ensure that the tasks are actually stopped in a timely manner, but is
off by default due
|         to HDFS-1208, where HDFS may respond to Thread.interrupt() by marking
nodes as dead.
|

```

```

|         If you run jobs in parallel, use :class:`pyspark.InheritableThread` for
thread
|         local inheritance.
|
|         See Also
|         -----
|         :meth:`SparkContext.cancelJobGroup`
|
|         Examples
|         -----
|         >>> import threading
|         >>> from time import sleep
|         >>> from pyspark import InheritableThread
|         >>> result = "Not Set"
|         >>> lock = threading.Lock()
|         >>> def map_func(x):
|         ...     sleep(100)
|         ...     raise RuntimeError("Task should have been cancelled")
|         >>> def start_job(x):
|         ...     global result
|         ...     try:
|         ...         sc.setJobGroup("job_to_cancel", "some description")
|         ...         result = sc.parallelize(range(x)).map(map_func).collect()
|         ...     except Exception as e:
|         ...         result = "Cancelled"
|         ...     lock.release()
|         >>> def stop_job():
|         ...     sleep(5)
|         ...     sc.cancelJobGroup("job_to_cancel")
|         >>> suppress = lock.acquire()
|         >>> suppress = InheritableThread(target=start_job, args=(10,)).start()
|         >>> suppress = InheritableThread(target=stop_job).start()
|         >>> suppress = lock.acquire()
|         >>> print(result)
|         Cancelled
|
|         setLocalProperty(self, key: str, value: str) -> None
|         Set a local property that affects jobs submitted from this thread, such
as the
|         Spark fair scheduler pool.
|
|         .. versionadded:: 1.0.0
|
|         Parameters
|         -----
|         key : str
|             The key of the local property to set.
|         value : str

```

```

|         The value of the local property to set.
|
|     See Also
|     -----
|     :meth:`SparkContext.getLocalProperty`
|
|     Notes
|     -----
|     If you run jobs in parallel, use :class:`pyspark.InheritableThread` for
thread
|     local inheritance.
|
|     setLogLevel(self, logLevel: str) -> None
|     Control our logLevel. This overrides any user-defined log settings.
|     Valid log levels include: ALL, DEBUG, ERROR, FATAL, INFO, OFF, TRACE,
WARN
|
|     .. versionadded:: 1.4.0
|
|     Parameters
|     -----
|     logLevel : str
|         The desired log level as a string.
|
|     Examples
|     -----
|     >>> sc.setLogLevel("WARN") # doctest :+SKIP
|
|     show_profiles(self) -> None
|     Print the profile stats to stdout
|
|     .. versionadded:: 1.2.0
|
|     See Also
|     -----
|     :meth:`SparkContext.dump_profiles`
|
|     sparkUser(self) -> str
|     Get SPARK_USER for user who is running SparkContext.
|
|     .. versionadded:: 1.0.0
|
|     statusTracker(self) -> pyspark.status.StatusTracker
|     Return :class:`StatusTracker` object
|
|     .. versionadded:: 1.4.0
|
|     stop(self) -> None

```

```

|         Shut down the :class:`SparkContext`.
|
|         .. versionadded:: 0.7.0
|
|     textFile(self, name: str, minPartitions: Optional[int] = None, use_unicode:
bool = True) -> pyspark.rdd.RDD[str]
|         Read a text file from HDFS, a local file system (available on all
|         nodes), or any Hadoop-supported file system URI, and return it as an
|         RDD of Strings. The text files must be encoded as UTF-8.
|
|         .. versionadded:: 0.7.0
|
|     Parameters
|     -----
|     name : str
|         directory to the input data files, the path can be comma separated
|         paths as a list of inputs
|     minPartitions : int, optional
|         suggested minimum number of partitions for the resulting RDD
|     use_unicode : bool, default True
|         If `use_unicode` is False, the strings will be kept as `str`
(encoding
|         as `utf-8`), which is faster and smaller than unicode.
|
|         .. versionadded:: 1.2.0
|
|     Returns
|     -----
|     :class:`RDD`
|         RDD representing text data from the file(s).
|
|     See Also
|     -----
|     :meth:`RDD.saveAsTextFile`
|     :meth:`SparkContext.wholeTextFiles`
|
|     Examples
|     -----
|     >>> import os
|     >>> import tempfile
|     >>> with tempfile.TemporaryDirectory() as d:
|     ...     path1 = os.path.join(d, "text1")
|     ...     path2 = os.path.join(d, "text2")
|     ...
|     ...     # Write a temporary text file
|     ...     sc.parallelize(["x", "y", "z"]).saveAsTextFile(path1)
|     ...
|     ...     # Write another temporary text file

```

```

|         ...     sc.parallelize(["aa", "bb", "cc"]).saveAsTextFile(path2)
|         ...
|         ...     # Load text file
|         ...     collected1 = sorted(sc.textFile(path1, 3).collect())
|         ...     collected2 = sorted(sc.textFile(path2, 4).collect())
|         ...
|         ...     # Load two text files together
|         ...     collected3 = sorted(sc.textFile('{}',{}'.format(path1, path2),
5).collect())
|
|         >>> collected1
|         ['x', 'y', 'z']
|         >>> collected2
|         ['aa', 'bb', 'cc']
|         >>> collected3
|         ['aa', 'bb', 'cc', 'x', 'y', 'z']
|
| union(self, rdds: List[pyspark.rdd.RDD[~T]]) -> pyspark.rdd.RDD[~T]
|     Build the union of a list of RDDs.
|
|     This supports unions() of RDDs with different serialized formats,
|     although this forces them to be reserialized using the default
|     serializer:
|
|     .. versionadded:: 0.7.0
|
|     See Also
|     -----
|     :meth:`RDD.union`
|
|     Examples
|     -----
|     >>> import os
|     >>> import tempfile
|     >>> with tempfile.TemporaryDirectory() as d:
|         ...     # generate a text RDD
|         ...     with open(os.path.join(d, "union-text.txt"), "w") as f:
|         ...         _ = f.write("Hello")
|         ...     text_rdd = sc.textFile(d)
|         ...
|         ...     # generate another RDD
|         ...     parallelized = sc.parallelize(["World!"])
|         ...
|         ...     unioned = sorted(sc.union([text_rdd, parallelized]).collect())
|
|     >>> unioned
|     ['Hello', 'World!']

```

```

|   wholeTextFiles(self, path: str, minPartitions: Optional[int] = None,
use_unicode: bool = True) -> pyspark.rdd.RDD[typing.Tuple[str, str]]
|       Read a directory of text files from HDFS, a local file system
|       (available on all nodes), or any Hadoop-supported file system
|       URI. Each file is read as a single record and returned in a
|       key-value pair, where the key is the path of each file, the
|       value is the content of each file.
|       The text files must be encoded as UTF-8.
|
|       .. versionadded:: 1.0.0
|
|       For example, if you have the following files:
|
|       .. code-block:: text
|
|           hdfs://a-hdfs-path/part-00000
|           hdfs://a-hdfs-path/part-00001
|           ...
|           hdfs://a-hdfs-path/part-nnnnn
|
|       Do ``rdd = sparkContext.wholeTextFiles("hdfs://a-hdfs-path")``,
|       then ``rdd`` contains:
|
|       .. code-block:: text
|
|           (a-hdfs-path/part-00000, its content)
|           (a-hdfs-path/part-00001, its content)
|           ...
|           (a-hdfs-path/part-nnnnn, its content)
|
|       Parameters
|       -----
|       path : str
|           directory to the input data files, the path can be comma separated
|           paths as a list of inputs
|       minPartitions : int, optional
|           suggested minimum number of partitions for the resulting RDD
|       use_unicode : bool, default True
|           If `use_unicode` is False, the strings will be kept as `str`
(encoding
|           as `utf-8`), which is faster and smaller than unicode.
|
|       .. versionadded:: 1.2.0
|
|       Returns
|       -----
|       :class:`RDD`
|           RDD representing path-content pairs from the file(s).

```

```

|
| Notes
| -----
| Small files are preferred, as each file will be loaded fully in memory.
|
| See Also
| -----
| :meth:`RDD.saveAsTextFile`
| :meth:`SparkContext.textFile`
|
| Examples
| -----
| >>> import os
| >>> import tempfile
| >>> with tempfile.TemporaryDirectory() as d:
| ...     # Write a temporary text file
| ...     with open(os.path.join(d, "1.txt"), "w") as f:
| ...         _ = f.write("123")
| ...
| ...     # Write another temporary text file
| ...     with open(os.path.join(d, "2.txt"), "w") as f:
| ...         _ = f.write("xyz")
| ...
| ...     collected = sorted(sc.wholeTextFiles(d).collect())
| >>> collected
| [('/.../1.txt', '123'), ('.../2.txt', 'xyz')]
|
| -----
| Class methods defined here:
|
| getOrCreate(conf: Optional[pyspark.conf.SparkConf] = None) -> 'SparkContext'
from builtins.type
|     Get or instantiate a :class:`SparkContext` and register it as a
singleton object.
|
| .. versionadded:: 1.4.0
|
| Parameters
| -----
| conf : :class:`SparkConf`, optional
|         :class:`SparkConf` that will be used for initialization of the
:class:`SparkContext`.
|
| Returns
| -----
| :class:`SparkContext`
|         current :class:`SparkContext`, or a new one if it wasn't created
before the function

```



```

|         call.
|
|     Examples
|     -----
|     >>> SparkContext.getOrCreate()
|     <SparkContext ...>
|
| setSystemProperty(key: str, value: str) -> None from builtins.type
|     Set a Java system property, such as `spark.executor.memory`. This must
|     be invoked before instantiating :class:`SparkContext`.
|
|     .. versionadded:: 0.9.0
|
|     Parameters
|     -----
|     key : str
|         The key of a new Java system property.
|     value : str
|         The value of a new Java system property.
|
|     -----
| Readonly properties defined here:
|
| applicationId
|     A unique identifier for the Spark application.
|     Its format depends on the scheduler implementation.
|
|     * in case of local spark app something like 'local-1433865536131'
|     * in case of YARN something like 'application_1433865536131_34483'
|
|     .. versionadded:: 1.5.0
|
|     Examples
|     -----
|     >>> sc.applicationId # doctest: +ELLIPSIS
|     'local-...'
|
| defaultMinPartitions
|     Default min number of partitions for Hadoop RDDs when not given by user
|
|     .. versionadded:: 1.1.0
|
|     Examples
|     -----
|     >>> sc.defaultMinPartitions > 0
|     True
|
| defaultParallelism

```

```

|      Default level of parallelism to use when not given by user (e.g. for
reduce tasks)
|
|      .. versionadded:: 0.7.0
|
|      Examples
|      -----
|      >>> sc.defaultParallelism > 0
|      True
|
|  listArchives
|      Returns a list of archive paths that are added to resources.
|
|      .. versionadded:: 3.4.0
|
|      See Also
|      -----
|      :meth:`SparkContext.addArchive`
|
|  listFiles
|      Returns a list of file paths that are added to resources.
|
|      .. versionadded:: 3.4.0
|
|      See Also
|      -----
|      :meth:`SparkContext.addFile`
|
|  resources
|      Return the resource information of this :class:`SparkContext`.
|      A resource could be a GPU, FPGA, etc.
|
|      .. versionadded:: 3.0.0
|
|  startTime
|      Return the epoch time when the :class:`SparkContext` was started.
|
|      .. versionadded:: 1.5.0
|
|      Examples
|      -----
|      >>> _ = sc.startTime
|
|  uiWebUrl
|      Return the URL of the SparkUI instance started by this
:class:`SparkContext`
|
|      .. versionadded:: 2.1.0

```

```

|
|     Notes
|     ----
|     When the web ui is disabled, e.g., by ``spark.ui.enabled`` set to
| ``False``,
|     it returns ``None``.
|
|     Examples
|     -----
|     >>> sc.uiWebUrl
|     'http://...'
|
| version
|     The version of Spark on which this application is running.
|
|     .. versionadded:: 1.1.0
|
|     Examples
|     -----
|     >>> _ = sc.version
|
| -----
| Data descriptors defined here:
|
| __dict__
|     dictionary for instance variables (if defined)
|
| __weakref__
|     list of weak references to the object (if defined)
|
| -----
| Data and other attributes defined here:
|
| PACKAGE_EXTENSIONS = ('.zip', '.egg', '.jar')
|
| __annotations__ = {'PACKAGE_EXTENSIONS': typing.Iterable[str], '_activ...

```

```
[4]: import json
```

```

fields = ['product_id',
          'user_id',
          'score',
          'time']
fields2 = ['product_id',
          'user_id',
          'review',

```

```

        'profile_name',
        'helpfulness',
        'score',
        'time']
fields3 = ['product_id',
          'user_id',
          'time']
fields4 = ['user_id',
          'score',
          'time']
def validate(line):
    for field in fields2:
        if field not in line: return False
    return True

reviews_raw = sc.textFile('data/movies.json')

reviews = reviews_raw.map(lambda line: json.loads(line)).filter(validate)

reviews.cache()

```

[4]: PythonRDD[6] at RDD at PythonRDD.scala:53

[5]: reviews.take(1)

[5]: [{'user_id': 'A141HP4LYPWMSR',
 'product_id': 'B003AI2VGA',
 'review': 'Synopsis: On the daily trek from Juarez, Mexico to El Paso, Texas an ever increasing number of female workers are found raped and murdered in the surrounding desert. Investigative reporter Karina Danes (Minnie Driver) arrives from Los Angeles to pursue the story and angers both the local police and the factory owners who employ the undocumented aliens with her pointed questions and relentless quest for the truth.

Her story goes nationwide when a young girl named Mariela (Ana Claudia Talancon) survives a vicious attack and walks out of the desert crediting the Blessed Virgin for her rescue. Her story is further enhanced when the "Wounds of Christ" (stigmata) appear in her palms. She also claims to have received a message of hope for the Virgin Mary and soon a fanatical movement forms around her to fight against the evil that holds such a stranglehold on the area.

Critique: Possessing a lifelong fascination with such esoteric matters as Catholic mysticism, miracles and the mysterious appearance of the stigmata, I was immediately attracted to the \'05 DVD release `Virgin of Juarez`. The film offers a rather unique storyline blending current socio-political concerns, the constant flow of Mexican migrant workers back and forth across the U.S./Mexican border and the traditional Catholic beliefs of the Hispanic population. I must say I was quite surprised by the unexpected route taken by the plot and the means and methods by which the

heavenly message unfolds.

`Virgin of Juarez\` is not a film that you would care to watch over and over again, but it was interesting enough to merit at least one viewing. Minnie Driver delivers a solid performance and Ana Claudia Talancon is perfect as the fragile and innocent visionary Mariela. Also starring Esai Morales and Angus Macfadyen (Braveheart).',

```
'summary': '"There Is So Much Darkness Now ~ Come For The Miracle"',
'profile_name': 'Brian E. Erland "Rainbow Sphinx"',
'helpfulness': '7/7',
'time': 1182729600,
'score': 3.0}]
```

```
[6]: num_movies = reviews.groupBy(lambda entry: entry['product_id']).count()
num_users = reviews.groupBy(lambda entry: entry['user_id']).count()
num_entries = reviews.count()

print (str(num_entries) + " reviews of " + str(num_movies) + " movies by " +
→str(num_users) + " different people.")
```

50000 reviews of 1539 movies by 36409 different people.

```
[7]: #Suggestion_users = reviews.filter(lambda entry: entry['user_id'])

#for review in Suggestion_users.collect():
r1 = reviews.map(lambda r: ((r['product_id'],), 1))
avg3 = r1.mapValues(lambda x: (x, 1)) \
→.reduceByKey(lambda x, y: (x[0] + y[0], x[1] +
→y[1]))

avg3 = avg3.filter(lambda x: x[1][1] > 20 )
avg3 = avg3.map(lambda x: ((x[1][0]+x[1][1],), x[0])) \
→.sortByKey(ascending=False)
```

```
[8]: for movie in avg3.take(10):
print ("http://www.amazon.com/dp/" + movie[1][0] + " WATCHED BY : " +
→str(movie[0][0]) + " PEOPLE")
```

http://www.amazon.com/dp/B002VL2PTU	WATCHED BY : 1338 PEOPLE
http://www.amazon.com/dp/B001NFMFMQ	WATCHED BY : 1296 PEOPLE
http://www.amazon.com/dp/B003DBEX6K	WATCHED BY : 1252 PEOPLE
http://www.amazon.com/dp/B000KKQNR0	WATCHED BY : 1216 PEOPLE
http://www.amazon.com/dp/B000ZLFALI	WATCHED BY : 1182 PEOPLE
http://www.amazon.com/dp/B000ZLFALS	WATCHED BY : 1182 PEOPLE
http://www.amazon.com/dp/B000063W82	WATCHED BY : 1178 PEOPLE
http://www.amazon.com/dp/B0001G6PZC	WATCHED BY : 1090 PEOPLE
http://www.amazon.com/dp/B0083SI986	WATCHED BY : 1062 PEOPLE
http://www.amazon.com/dp/B0014ERK00	WATCHED BY : 1030 PEOPLE

```
[9]: r2 = reviews.map(lambda ru: ((ru['user_id'],), 1))
avg2 = r2.mapValues(lambda x: (x, 1)) \
        .reduceByKey(lambda x, y: (x[0] + y[0], x[1] +
        ↪ y[1]))

avg2 = avg2.filter(lambda x: x[1][1] > 20 )
avg2 = avg2.map(lambda x: ((x[1][0]+x[1][1],), x[0])) \
        .sortByKey(ascending=False )
```

```
[10]: for movie in avg2.take(10):
        print ("http://www.amazon.com/dp/" + movie[1][0] + "    WATCHED : " +
        ↪str(movie[0][0]) + " MOVIES")
```

```
http://www.amazon.com/dp/A3LZGLA88KOLAO    WATCHED : 112 MOVIES
http://www.amazon.com/dp/ANCOMAI0I7LVG    WATCHED : 108 MOVIES
http://www.amazon.com/dp/A2NJO6YE954DBH    WATCHED : 108 MOVIES
http://www.amazon.com/dp/A328S9RN3U5M68    WATCHED : 106 MOVIES
http://www.amazon.com/dp/A16CZRQL23NOIW    WATCHED : 102 MOVIES
http://www.amazon.com/dp/A100DC971MDHV8    WATCHED : 100 MOVIES
http://www.amazon.com/dp/A35ZK3M8L9JUPX    WATCHED : 100 MOVIES
http://www.amazon.com/dp/AB02ZI2Y5DQ9T    WATCHED : 100 MOVIES
http://www.amazon.com/dp/A1GG0C9PVDXW7Z    WATCHED : 98 MOVIES
http://www.amazon.com/dp/A7Y6AVS576M03    WATCHED : 94 MOVIES
```

```
[11]: # Has someone written a review?
filtered = reviews.filter(lambda entry: "George" in entry['profile_name'])
print ("Found " + str(filtered.count()) + " entries.\n")

for review in filtered.collect():
    print ("Rating: " + str(review['score']) + " and helpfulness: " +
    ↪review['helpfulness'])
    print ("http://www.amazon.com/dp/" + review['product_id'])
    print (review['summary'])
    print (review['review'])
    print ("\n")
```

Found 102 entries.

Rating: 4.0 and helpfulness: 1/1

http://www.amazon.com/dp/B000NDFLWG

In Debt We Trust

Very enlightening on the development of the debt problems we all face especially how we have gotten there with many ideas as to how to decrease our debts and change our purchasing patterns. All families and individuals should view this, an eye opener.

Rating: 5.0 and helpfulness: 10/13

<http://www.amazon.com/dp/B0001G6PZC>

My Favortite movie by far

Way better than Lord of the Rings. The battle sequences were suberb and stunning. I loved the fighting and plot. Tom Cruise was awesome in acting in this movie. It was also historicly accurate. So you can learn from this film. It was very educational. It deserves some academy awards. If you like movies like Gladiator you'd pamper yourself by coming to this movie

Rating: 5.0 and helpfulness: 3/5

<http://www.amazon.com/dp/B0001G6PZC>

Epic movie in the traditions of "Braveheart"

Wow, what a movie! First of all, the story line is so intersting. It is around 1876 and Japan is on course to introduce western lifestyle in the country. However, the Samurais, the traditional Japanese warriors they do not want to except this novice. Thus emperor of Japan confront them military. In order to train the Japanese military the US army captain (portrayed by Tom Cruise) is invited. During one of the confrontations he is taken as a prisoner by Samurais. He spends in Samurai village several month. He studies their lifestyle, their traditions and realizes that he was on the wrong side of the battlefield. Thus he decides to stand next to the Samurais and fight with them for their freedom.

This is an epic movie. It probably can be compared to "Braveheart". Tom Cruise is brilliant, his acting is amazing. This is must have for movie lovers. I still do not believe that Oscar committee prefered "The Lord of the Rings" to "The Last Samurai". This is really dissapointing.

Rating: 4.0 and helpfulness: 0/1

<http://www.amazon.com/dp/B0020HDRF2>

Terminator: Target

Taking the advice from some people here, I found the 2-disc with a few extras at Target. Apparently Target has the exclusive deal on this. The movie is the "Director's Cut" with slight variations to the theatrical version. There is no commentary track here. The Special Features are found on the second disc along with the digital copy. Personally, the digital copy doesn't do a thing for me. First up is "Reforging The Furture" which runs 18:59 and shows how the filmmakers reinvented the franchise. Then there are 4 featurettes - "Digital Destruction", "Enlisting The AirForce", "An Icon Returns", "Stan Winston Shop". These four total 11:30. So is it worth it to spend the extra cash for just a hand full of extras? I guess, but I would have loved to have more extras including a commentary.

Rating: 5.0 and helpfulness: 7/10

<http://www.amazon.com/dp/6300147967>

Faith worth rescuing

When I first saw this movie it lifted me up. If only a real Pope could be so spiritual. Lent it to a Buddhist priest and a Catholic friend and they both

praised the movie, and the feelings it brought out in them.

The Giannini character says God doesn't exist but he hopes that's not true. The movie makes you believe that, if God does exist, it would be a wonderful thing. And maybe God does exist.

But it's not a religious movie, it's a movie about people and a movie that's wonderfully done . . "Full of wonder" . . hmm.

Rating: 5.0 and helpfulness: 0/0

<http://www.amazon.com/dp/B00004CQTP>

Killer!

I would have to disagree that this is better than the original. I'm not saying that it is bad, because that would be blasphemy, but the original still remains my favorite.

Rating: 5.0 and helpfulness: 2/6

<http://www.amazon.com/dp/0800103688>

cameron diaz as superb as always, roberts' worst movie

julia roberts is not good at all, but diaz is perfect even if her musical skills are not revealed. the story is funny, so buy this movie

Rating: 5.0 and helpfulness: 0/0

<http://www.amazon.com/dp/B000063W82>

the only way i could remember

i'm 50 and dropped out of school in the 8th grade. But i know how the US government works and what the Preamble to the United States Constitution is, and all the things that this dvd has on it. and i learned it saturday morning when i was a kid, waiting for the next cartoon to start. if your kid is a straight A student he won't need this. get him boxing lessons he'll probably need them, but if he's a idiot like me this will Help him in his Future when he go's back to school after he joins the Army, gets married and has 4 kids and wakes up one morning and says to him self. "how in the hell am i going to feed this bunch without a education". the info he accidentally learns on this dvd will give him the Confidence to give school 1 more shot, and make it this time. good luck

Rating: 5.0 and helpfulness: 5/5

<http://www.amazon.com/dp/6303955746>

Repertory of pieces

This extraordinary video lacks about a relationship of the repertory of pieces performed by Maestro Andres Segovia. I detail them next for all those that are interested:<p>(1)-LA MAJA DE GOYA, by Enrique Granados (arrangement Miguel Llobet).<p>(2)-SARABANDE from Lute Suite I, BWV 996, by J.S.Bach.<p>(3)-GAVOTTE EN RONDEAU, from Lute Suite IV, BWV1006A, and Violin Partita III BWV 1006, by J.S.Bach.<p>(4)-FANDANGUILLO from 'Suite Castellana', by F.M.Torroba.<p>(5)-LA FILLA DEL MARXANT ('The Merchant's Daughter'), a popular Catalanian song,

arranged by Miguel Llobet.<p>(6)-STUDY Lesson 26 from 'The Complete Works for Guitar' Vol.2 page 52 (Chanterelle 802), by Dionisio Aguado.<p>(7)-RECUERDOS DE LA ALHAMBRA (music in background) by Francisco Tarrega.<p>(8)-LA ARRULLADORA, Lullaby from Suite 'Platero and I' by Mario Castelnuovo-Tedesco.<p>(9)-SPANISH DANCE Nr.10-in G by Enrique Granados (arranged by Miguel Llobet).<p>(10)-LA MAJA DE GOYA (music in background), by Enrique Granados.<p>It would be very interesting, for all the fans to the classical guitar, could provide all Videos of Andres Segovia in DVD format.

Rating: 4.0 and helpfulness: 0/0

<http://www.amazon.com/dp/B00000HF14>

The action heavy part of the Tenchi Muyo OAVs

This particular volume in the Tenchi Muyo OAVs brings us the conflict with Kagato, the introduction of another girl into Tenchi's extended family and a comedy episode with some foreshadowing for things to come. It still features the usual mix of comedy and plot twists galore that the Tenchi Muyo OAVs are well known for. The disc comes with Japanese and English audio which will satisfy fans of both preferences. But special features in this disc are lacking as the only notable feature is a collection of trailers for titles that Pioneer/Geneon recently distributed and licensed back in 2002/ 2003 when this DVD first came out. If you just want the episodes, then this volume of the Tenchi Muyo OAVs is worth purchasing. If you want special features, then try hunting down the Tenchi Muyo Ultimate Edition DVD set which has an extensive offering of special features for fans to go giddy over.

Rating: 4.0 and helpfulness: 0/1

<http://www.amazon.com/dp/B004EPYZQ2>

George in Virginia

I think the hype before the movie set my expectations higher than they should have been. This movie is Speilburg's version renewed of E.T. and a less threatening isolated version of alien invasion from a military controlled Area 51 escape. The child imagination will enjoy this one but the adult imagination will stop believing the events are possible by the end of the movie.

Rating: 3.0 and helpfulness: 1/1

<http://www.amazon.com/dp/B000FVQLQQ>

Overrated...yes

I saw this movie for two reasons. I didn't think I would have to tolerate car chases and bomb blasts for two hours and the film, to be sure, relies on neither. However, I can't help but wonder why people are calling this a great movie. Although I like Mr. Crowe as an actor, his work in this film seems overly self conscious and his accent wears on you by the end of the film. The one bright star is the female lead. She is a wonderful actress and I hope to see more from her. If you want to see a truly great film get "Remains of the Day". Hopkings and Thompson define great acting in this film...

Rating: 5.0 and helpfulness: 0/0

<http://www.amazon.com/dp/B002LSIAQ0>

a story of a security guard who works on our USS North Carolina
stories of our Navy people who give thier lives to defend our Constitution and
our way of life we so love, who still seem to be on our USS North Carolina which
is on display down on our coast

Rating: 5.0 and helpfulness: 0/1

<http://www.amazon.com/dp/B000065K9R>

I watched Scary Movie last Summer and I screamed!!

Scary Movie looked good from the moment I laid my eyes on it. As a teen slasher
fan I loved the humor and could recognize all the parts from other films. The
characters are totally clueless and some what stupid but that makes it more
funny. Some of the jokes are a killer!! Scary Movie is just a hilarious
rollercoaster ride of fun and mayhem. I've been waiting for a spoof like this
and I got one!! If you want to laugh until you cry watch Scary Movie!

Rating: 5.0 and helpfulness: 0/0

<http://www.amazon.com/dp/B004EPYZQM>

Awesome!!

Okay, I love racing movies, especially the Fast and Furious movies with their
constant racing, girls, and girls. Yeah, I said it. Anyways, I've always been a
fan of Vin Diesel and Paul Walker. They are both great actors, along with making
the movie a bit more exciting with their, on and off relation ship of deciding
weather he was a part of the family or not. This movie, had so much actions, I
actually sat through the whole thing, and ate my popcorn. I recommend it, watch
it now! D:

Rating: 5.0 and helpfulness: 5/5

<http://www.amazon.com/dp/B000NVL49W>

A superb production

This must be one of the best DVDs of opera available.
Although it is old
(1974) the image quality is more than fair, being a studio production The sound
quality is excellent.

The director, Götz Friedrich, really
exhibits his genius in this production. The sets are wonderful and very
atmospheric, with an immense well at the side of the stage and the courtyard of
Herod's palace being as it should. The costumes are very good, but above all the
acting and the movement of the principles is superb.

Teresa Stratas
must have given the performance of her life. She is perfect as Salome, playing
the part of spoiled little brat that always had to have her own way, ignoring
and really not caring about the consequences. She moves so much in tune with
Strauss's music, she acts superbly and sings beautifully. Astrid Varnay,
although grotesque in appearance portrays superbly the part of her mother. Bernd

Weikl is vocally excellent and acts very well as Jochanaan, but he appears rather plumb and overfed for a starving prophet. The rest of the cast is good and Hanna Schwarz, as usual, is an asset to any production.
Karl Böhm and the Vienna Philharmonic provide an immensely rich orchestral sound with a very tight and dramatic conducting from Böhm.

I have seen other productions of Salome on DVD and on the stage. I firmly believe that none of them even begins to approach this one in terms of quality. I recommend this set as being far better than all the others, including Peter Hall's superb production from Covent Garden.
R. Strauss - Salome

Rating: 5.0 and helpfulness: 3/3

<http://www.amazon.com/dp/5556167281>

Supercalifragilisticxpealidocious --- Most Delightful Way!

Come along now, children, let's read this review. No straggling now. There you go.

I must have only seen "Mary Poppins" when I was a little kid because I only barely remembered the movie when I was watching it and definitely didn't remember the plot. Definitely an experience worth having, whether you've seen it once, or many, many times.

The basic plot is that two kids have been a real terror to their nannies, but not so much out of meanness as the nannies' domineering personalities. They draw up a list of their needs for a nanny and when the wind changes, Mary Poppins comes, despite their tight-laced fathers' wishes. They're soon off on fantastical adventures that may just change the lifestyle of their entire family! This movie makes important comments about suffrage, structure, and just finding the joy in life. It's supercalifragilisticxpealidocious! (And I didn't even have to look that up!)

The movie is presented in wide-screen with fully-restored, bright and colorful animation and practically perfect sound. There's also both a subtitle and a close-caption track, although oddly these differ greatly in what they do and don't pick up on. I ended up keeping on both.

Like other recent Disney Platinum releases, this DVD contains a wealth of special features that will help you to enjoy the movie more. Chief among them is a great deleted song. You also get a complete making-of feature, a look at the movie magic, a game, promo posters and trailers and many other simply delightful goodies.

Like the "Aladdin" release, this release is loaded with both a commentary track and pop-up fun-facts. If you actually decide to watch all of this stuff, you'll end up having seen the film at least three times by the time you're done!

So what are you waiting for? Go off now and buy this! Spit-spot!

Rating: 5.0 and helpfulness: 0/4

<http://www.amazon.com/dp/B000I9WVZA>

black humor can be funny(ach, german word say "ar")

This movie would have not been half as funny without Cameron Diaz. I Love Cameron's ideas for the wedding(like John-John Kennedy), She's very convincing. But there is a little too much dead people, God bless them, thank

God, Diaz doesn't die(ouffffff)

Rating: 3.0 and helpfulness: 4/5

<http://www.amazon.com/dp/B000VHVG8>

a poor transfer to blu-ray, for a great film

It is extremely annoying when a title is re-released in Blu-ray format with no restoration, or for that matter with no improvement on the image quality provided by the original DVD. Most films placed on Blu-ray stand out because of their crystal clear image and excellent sound. This makes it worth buying again, even if one possesses the original DVD copy, simply to have the improved performance on the screen, which may be even further enhanced by good quality equipment. This Blu-ray edition is far inferior to the original DVD edition, which used to have a number of interesting extras and the Blu-ray has absolutely none!

As to the film itself, in my opinion, it still rates as one of the best comedies ever to be produced for the cinema. A very clever story, by John Cheese, with tremendous dialogue and very good acting from all the principals. There are some exaggerations in the plot, but this is true of most comedies.

I noticed that the Blu-ray edition of this film will be released in the UK in August. I hope that after this long delay, this release will have the necessary improvements, especially in image quality, to make it worth a 5 star rating, and possibly induce one to acquire once more.

Rating: 5.0 and helpfulness: 1/1

<http://www.amazon.com/dp/B000A9QK8M>

The Mark of Zorro

I've always enjoyed this type of movie. The fencing scene between Tyrone Power and Basil Rathbone was, and still is one of the best action scenes in all movies. Rathbone was an accomplished user of the foil and was to have said, "Tyrone Power could have fenced Errol Flynn into a box." I am in awe at times of the athletic abilities that some actors have. Great movie, one of Hollywood's finest as far as I'm concerned. I highly recommend it for all audiences. It is an example of a movie made without the electronic wizardry that prevails today

Rating: 4.0 and helpfulness: 0/0

<http://www.amazon.com/dp/B000AMWIVM>

See SAW

My corny title is not just there to be funny in a stupid way, as it is also meant to convey to horror fans who have not seen SAW, thinking it to be another HOSTEL/HILLS HAVE EYES/any "let's torture young people for fun instead of trying to be, you know, scary" movie, that it's definitely worth their time. I have not seen SAW's sequels (though I liked this one enough to see the others), but I can say that this movie is more in the SE7EN kind of vein than the torture porn flicks we've been subjected to over the last few years. Yes, it is quite often gruesome, graphic, and disturbing, but it is all accredited to a very clever and

darkly interesting premise.
We are introduced to our two protagonists (if they can be called that, as they have their vices) first, workaholic doctor Lawrence (Cary Elwes) and cocky youngster Adam (screenwriter Leigh Whannell), who awaken in a dark room resembling a larger version of a prison cell, each of them chained to pipes adjacent from each other. In the center is a dead body holding an unloaded gun with a gunshot wound to the head and a tape recorder. Adam and Lawrence each have tapes to be played on that recorder, on which a raspy voice, their captor, informs Lawrence that he must kill Adam by 6:00 that night, or he will kill Lawrence's wife and daughter. To further explain what could possibly be going on, Adam and Lawrence both tell certain stories about their past, leading to flashbacks where we are introduced to other key characters like policeman Det. David Tapp (Danny Glover) and his partner, Det. Steven Sing (Ken Leung), who are on the trail of the Jigsaw Killer, a sadistic serial killer who forces his victims to save their own lives through twisted puzzles and games. Lawrence believes this Jigsaw is their holder, and that Tapp believes Lawrence is somehow in cahoots with Jigsaw. More plot complications pile on top of each other, leading to the bloody, shocking finale, which whether you like it or not, is a doozy.
Writer Whannell and director James Wan, both newcomers, showcase a great confidence towards filmmaking, as Whannell's writing is very fluid and engrossing (I'm not saying he's David Mamet, but for someone not much older than college age he's very talented) and Wan's direction points toward a long future in filmmaking. Elwes should have won some sort of horror acting award or at least award for best desperate screaming and wailing (something I can't quite say for Whannell, a much better writer than he is an actor, though he's more than adequate in his role). I can't tell you if the sequels are able to have a nice psychological thriller along with the blood and guts, but this one has it and more.

Rating: 5.0 and helpfulness: 1/1

<http://www.amazon.com/dp/B000ANVPPQ>

O'Hara and Wayne at their BEST

I finally own the movie that almost out does the Quiet Man. If you love O'Hara and Wayne together, you cannot miss this one. A thrilling comedy for the whole family for now and years to come. You will laugh till it hurts and watch it twice because you will miss a lot while laughing out loud. Amazon delivers an excellent quality DVD.

Rating: 4.0 and helpfulness: 2/2

<http://www.amazon.com/dp/B001IDSZ10>

Where Are The Rest?!!

This is a great escapist revenge film for anyone who's seen the evening news and been disgusted with the bleeding heart "solutions" to crime & punishment. One question though, where are the DVD versions of Deathwish 2-4?

Rating: 5.0 and helpfulness: 2/3

<http://www.amazon.com/dp/B003PX4S9I>

Outstanding Series!

It was very enjoyable seeing these now forty year old classics. Dean Martin was his usual easy going self as Matt Helm, while James Gregory was excellent as McDonald his boss. Well worth it!

Neal Robertson

Rating: 4.0 and helpfulness: 0/0

<http://www.amazon.com/dp/B000X9JZVI>

Mediocre quality but enjoyable

As a social studies teacher, I found this film to be both entertaining and full of information. Finally, we teachers could no longer get it The film copies became repeatedly broken, spliced, and broken again. I was happy when I was able to get it on DVD. The DVD was obviously copied from a low quality film and the DVD (especially the sound) is thus mediocre. However, I love that film and the feeling it gives me when I watch it, so I am happy to have it.

Rating: 3.0 and helpfulness: 2/2

<http://www.amazon.com/dp/B0001ADAVK>

I Like This Movie for Reasons OTHER than What Was Billed.

Mona Lisa smile. Oh, goodness, what a trip. Having gone to an all-girls high school in New England that has always been a feeder for Wellesley (and the Wellesley of today is quite different than in the 50's), I was admittedly curious about the film. However, all my good sense told me to stay away. I had enough feminism shoved down my throat at that school to last me a lifetime. <p>That said, I eventually saw the film as a captive audience member on an overseas flight. As they jumped in to the story, all the stereotypes came screaming out of the screen to me: stuffy old New England school, liberal from California, jaded students who know how to act the part for teachers, rich kids with parents on trustee boards, etc. I think the movie relies on these well-worn characters because the writers certainly didn't do a very good job of developing them. You just sort of fall into a familiar world.<p>My main criticisms of the film are:
1. As the movie progressed, I kept thinking that the storyline wasn't in sync with the character development. They kept jumping ahead in the plot, assuming that you are keeping up with all the nuances of all the girls as they bloom in their senior year. Really, this wasn't too hard as they kept the girls pretty one-dimensional. <p>2. The whole 'Mona Lisa' concept was forced and ridiculous. It didn't make sense that Julia Robert's lover called her that, and Kirsten Dunst's scene where she discusses this with her mother is laughable. You knew what Dunst was getting at, but the film must have relied heavily on the audience's personal sentiment because they certainly didn't bring you there emotionally. <p>3. And the ending...my goodness what a rip-off of 'Dead Poet's Society.' Many movies have ripped off 'Dead Poet's Society,' but none seemed as clear as this. I won't give it away, but don't worry...the girls didn't all stand up on their desks at least.<p>For all that, I don't think the movie was a complete failure. I thought the following were noteworthy in the film:
1. First off, I need to praise Maggie Gyllenhaal's

acting!!! 'Who?' you may ask. She was the 4th one mentioned in all the previews and was the one that no-one really knew. Her character could in some ways be a tired cliché, but she brought it to life and easily stole every scene she was in. I was really mesmerized by her performance, and I look forward to more in the future.<p>2. Ginnifer Goodwin (the cello player) and Marcia Gay Harden (the old spinster) also turned very solid performances.<p>3. The fact that Julia Roberts' character was ultimately able to turn down relationships with 2 charming men was fairly remarkable. The fact that she had, needed and enjoyed these relationships so much shows that she is not anti-male or anti-marriage. She merely has a mature understanding that marriage is not an absolute good and that neither of these men could ultimately stand by her.<p>4. What they did with Julia Styles' character (girl who was accepted into Yale Law School) was also impressive in this 'progressive' film. That character perhaps single-handedly kept this from being just empty feminist propaganda.<p>5. Kirsten Dunst's role (the girl who gets married) & her relationship with her mother gives some spooky insight into 'advice' some girls were getting from their mothers. The night her mother turns her away from her home is particularly painful.<p>Overall, I think the film is pretentious and completely misses what it aimed to be. Additionally, I don't think Roberts, Dunst or Styles turned in particularly outstanding performances. However, there was enough life there from random sources to keep the movie afloat.

Rating: 4.0 and helpfulness: 2/3

<http://www.amazon.com/dp/B0001ADAVK>

Mona Lisa is definitely smiling

Mona Lisa Smile is an inspiring film based on a time period when women are beginning to explore alternative roles in society other than the traditional roles of being a stay at home mother who concentrates on her husband's career and success--rather than her own. Julia Roberts and her fellow actresses explore the roles of women as Julia plays an art professor who sets out to academically enlighten a group of already well-educated, rich girls attending a finishing school. Through this enlightening process she helps several of the students find their way, broaden their horizons, and in the process finds herself. The movie deserves four stars because Julia's character not only lets the audience relate on a feminine level but on a humanitarian level as well. The viewer can truly imagine the Mona Lisa smiling, if she were to watch Julia Roberts' performance in this movie.

Rating: 5.0 and helpfulness: 0/0

<http://www.amazon.com/dp/B009D0AZIO>

Seen it five time and will see it again

I saw it at it's Festival premier in July 2011 and fell in love with the film and became a Polly Parton fan. Since then I've seen it at least 5 times it still makes me cry and laugh. It's a simple premise, deliver a package from point a to point b. Throw in two great guys (the twins), the people they know (in Hollywood) and the people they meet (on the road), a dream, and a super star,

you come out with a must see and must own film.

Rating: 4.0 and helpfulness: 2/8

<http://www.amazon.com/dp/B00004WM3D>

Good Movie, Good DVD

****This Review is only for the special edition DVD****
The extras on this DVD are not so good, the extras include: High Definition Film Transfer, the original opening Bank Robbery deleted scene, "Return to Escape from New York featurette, and 2 full length audio commentaries.
I think the only REAL extra on this DVD is that it includes a Mini Comic Book titled "The Snake Plissken Chronicles and cool packaging. Good DVD if you are a DVD collector or like the movie. Like it ALOT.

Rating: 3.0 and helpfulness: 1/1

<http://www.amazon.com/dp/B000VII4D6>

Perfect Stranger

This movie at first was starting to evaluate the stars, afterwards the rest of movie was thrilling and suspenseful.
Halle Berry very good actress.

Rating: 1.0 and helpfulness: 4/12

<http://www.amazon.com/dp/B0002EXFNI>

Pretty lame Movie about Nerd getting babe

Porn star is the girl next door, as it turns out. This movie sells for the same reason that the girls who tease get what they want. While there is some humor to this story, that's about all that I can say that is good about this film.

Rating: 5.0 and helpfulness: 1/1

<http://www.amazon.com/dp/B00008G1PX>

An excellent release of an old favorite

One of Disney's crown achievements. This version is greatly enhanced. It is a pleasure to watch, and the special features, including live interviews of the stars, make this a must have for any collection.

Rating: 5.0 and helpfulness: 1/1

<http://www.amazon.com/dp/B00008G1PX>

Night and Day, Watch Cinderella!

One of Disney's classic films, "Cinderella" is now available on this two-disc Platinum Edition DVD set. It's a tale of a girl who has a dream and finds that though she's been dealt a hard lot in life, her dream can still come true.
I think pretty much anyone that would be reading this would know this story, but just in case --- "Cinderella" features a girl, Cinderella (of course!), her two wicked stepsisters and her cruel and evil stepmother. Also

living with her are a spoiled, mean-hearted cat, Lucifer and a group of cute little mice who can talk to her and she can talk to. Things weren't always like this for Cinderella. Her father was a kind man and things had always been nice for her, but when he passed away, her stepmother revealed her true colors. She took the family fortune for her and her daughters and made Cinderella a servant girl, forced to tend to their every whim. All of this has not made Cinderella bitter, though, and her heart and spirit remain as pure and beautiful as the day she was born.

Now, things are starting to look up for Cinderella. She's made a new friend, a new mouse in the house and a royal ball has been announced. She really wants to go and is told that she can, if she has a lovely dress to wear. But when the wicked stepsisters ruin the dress, even she can't help but begin to despair... until a little magic happens.

"Cindrella" is a powerful tale from beginning to end. The viewer gets involved not just with the story, but with the characters as well, even the affairs of the smallest characters of all, the mice. The story is moved along by several great songs including "A Dream Is a Wish Your Heart Makes," "Bibbidi Bobbidi Boo" and "So This is Love."

This new Platinum Edition DVD set is a great way to rediscover the Disney "Cinderella" story. It features a digital presentation with enhanced picture and sound, 5.1 Surround Sound Theater Mix and French and Spanish Language tracks and subtitles.

Of course, this wouldn't be a Platinum Edition DVD if it didn't have great special features. My favorite is "Cinderella and Perry Como," a black-and-white segment featuring the live-action Cinderella, the voices of the mice and Perry Como. They talk about the movie and Perry Como expertly performs "Bibbidi Bobbidi Boo" along with the voices of the mice. There are several deleted songs, my personal favorite being the "Cinderella Work Song." This is an alternate take on the song performed in the movie by the mice as they make their dress for Cinderella. Cinderella fantasizes about what it would be like if there are more of her.

Other bonuses include publicity pictures and characters sketches, a short segment from "The Mickey Mouse Club," a "Laugh-o-Gram" that's a simplified version of the Cinderella story and the promos for the original theatrical release and subsequent video releases and theatrical rerelease. There are also games, including a Princess Pajama Party.

Perhaps the most unusual special feature is the "Cinderella Stories." Apparently hoping to lure sports fans, or get girls' fathers interested, these are ESPN segments about various sports teams that won great victories against tough odds.

All in all, a great package. I'm now interested to see the upcoming Cinderella III, for which a short trailer was included in this release. Apparently the Stepmother gains some sort of black magic power and takes Cinderella's happily-ever-after away. Harsh!

Rating: 5.0 and helpfulness: 0/1

<http://www.amazon.com/dp/B00008G1PX>

Memorable Film Indeed

What I remember (years after seeing it in theatre, and long after buying video for older nieces), was how pleasant Cinderella was to stepsisters Drisella and Anastasia however mean they were to
her. She said pleasantly "Good

Morning, Drisella (Anastasia).
Sleep well?" when bringing breakfast trays.

 Most notably, I remember that music lesson that the step-
mother gave the stepsisters. Stepmother played harpsichord; Drisella sang "Oh Sing Sweet Nightingale" (sounded awful); Ana-
stasia played the flute (didn't sound pretty either, maybe in wrong key). Then Drisella grabbed the flute and bonked poor Ana-
stasia! <p> These scenes revealed much about the three girls, which is much of the beauty of the film.

Rating: 5.0 and helpfulness: 0/0

<http://www.amazon.com/dp/B002YJMMBA>

Another awesome adaptation from Ron Howard

I was fairly surprised by this movie, it is a lot less dry then The Da Vinci Code was. The plot was well thought out and the amount of research Dan Brown put into the book clearly translates to the movie in a very interesting way. I thought the plot was pretty good and it kept me on my toes right until the end. The "bad guy" aspect of the movie really threw me for a loop as I had always assumed the guy with the glasses that was killing the Cardinals was the one they were after, but the ending was shocking when we learn who was really behind it all.

I am not religious at all, I am an atheist actually, but I thought this video showed a side of the church that not many people get to see and also made some very appropriate points in its message. The explosion sequence was absolutely amazing, just hope we never really find out what an antimatter explosion would look like for real.

Rating: 5.0 and helpfulness: 1/1

<http://www.amazon.com/dp/B00005M1Y7>

GREAT learning tool for grandparents to use

The Aristocats (Disney Gold Classic Collection) The original version of Aristocats is simply wonderful. We have watched it over and over with our grandchildren. It is fun for them and fun for us. They nearly always find something new in the very elaborate drawings. The voices are marvelous, the story is delightful and there is just enough suspense for young children. We cannot recommend this version enough. It makes one realize how bland the Disney stories are--story, drawing, even the animation is far more sophisticated in this old version. Jane and George Webb

Rating: 4.0 and helpfulness: 0/1

<http://www.amazon.com/dp/B001FB55HQ>

Terminator: Target

Taking the advice from some people here, I found the 2-disc with a few extras at Target. Apparently Target has the exclusive deal on this. The movie is the "Director's Cut" with slight variations to the theatrical version. There is no commentary track here. The Special Features are found on the second disc along with the digital copy. Personally, the digital copy doesn't do a thing for me.

First up is "Reforging The Furture" which runs 18:59 and shows how the filmmakers reinvented the franchise. Then there are 4 featurettes - "Digital Destruction", "Enlisting The AirForce", "An Icon Returns", "Stan Winston Shop". These four total 11:30. So is it worth it to spend the extra cash for just a hand full of extras? I guess, but I would have loved to have more extras including a commentary.

Rating: 4.0 and helpfulness: 0/0

<http://www.amazon.com/dp/B000GETWF4>

If only it was longer!

The first season of The Office does not disappoint, except in its length. Not having paid attention closely, I was surprised to find that it was only six episodes. That aside, the six episodes were all extremely entertaining. Steve Carrell is ingenious as Michael Scott, the regional manager with absolutely no class. He really shines in "Diversity Day." Also, as someone who started watching The Office during the second season, it was nice to be able to come back and see how everything began. This was definitely a worthwhile purchase!

Rating: 4.0 and helpfulness: 0/0

<http://www.amazon.com/dp/B000GETWF4>

Hilarious

Awesome show. Love the product. Six episodes should NOT cost this much. But it's good to have it.

Rating: 4.0 and helpfulness: 7/7

<http://www.amazon.com/dp/B000GETWF4>

Terrific Show Deserves Better DVD Package

The problem with "The Office: Season One" is not with the show itself. Steve Carell, who broke through big-time in 2005 with not only this series but his feature film "The 40 Year Old Virgin", leads a fine ensemble cast in this documentary style comedy about the daily business practices of a Pennsylvania-based paper company. It's a witty, frequently funny, and sometimes painfully honest look at the way a boss treats his employees. No, the problem once again lies with the DVD's distributor, Universal. This is a brand new show, yet the episodes on the DVD seem to have a murky, bland appearance, unlike the episodes which are seen weekly on NBC. I complained about this problem with the "Miami Vice: Season One" DVD, but at least you can justify that somewhat because the episodes were from twenty years ago. Universal, what are you DOING? Please take the time and effort to put out a better DVD set. Of the extras available, there are a generous amount of deleted scenes from each of Season One's six episodes, but no bloopers, which is a disappointment. I don't know if Universal actually listens to their customers, but when I go out to buy Season Two of this excellent show on DVD, they'd better show some marked improvement. The fans deserve better.

Rating: 5.0 and helpfulness: 20/20

<http://www.amazon.com/dp/B001JPC4N6>

Nothing Funnier

Nothing Sacred is often referred to as a screwball comedy. Granted it has Carole Lombard the ultimate screwball heroine and other elements which suggest this form of comedy. However, it does not in the end so much resemble films like Bringing up Baby as it anticipates the later satires of Preston Sturges. Whichever way it is classified though, Nothing Sacred remains one of the best comedies of the thirties or indeed of any era. <p>This film, with its story of an apparently dying young woman, has an added poignancy for viewers watching it today. Lombard would be dead by 1942 and a real national heroine also with a navy warship named after her. The fiction at times seems to look ahead to the fact and thus the story becomes moving as well as funny. <p>The print of the film on this Lumivision DVD is on the whole good. There are some scenes especially at the beginning of the film where the early Technicolor print has been damaged. This means that there is some flickering and some separation of the colours. For the most part however, the film looks almost perfect with the vibrant colours so typical of thirties Technicolor. It is a joy to see Carole Lombard in colour and something of a surprise also, for her hair looks redder than expected. There are one of two slight jumps where the continuity of the film has been broken and this may account for the running time of the film, at 73 minutes, being slightly shorter than that listed in some sources. <p>The extras on this DVD are welcome but rather disappointing. The two Sennett short comedies are not very funny and really show that he had not moved on at all from his early pioneering days. These shorts are of interest mainly because they allow us to see a young and barely recognizable Lombard toiling away in her pre-stardom days. The DVD also includes a very damaged trailer for Nothing Sacred and some interesting if rather brief home movies with Lombard and Gable.

Rating: 5.0 and helpfulness: 0/0

<http://www.amazon.com/dp/B001ILFUDM>

A classic for young and old

The Two-Disc 70th Anniversary Platinum Edition was a great hit for my 11 year old granddaughter. She has watched it several times in the last week. It's a great buy and gift. Masterfully reproduced on disc.

Rating: 5.0 and helpfulness: 1/1

<http://www.amazon.com/dp/B001ILFUDW>

The Perfect Teaching Tool

Our grankids love this musical video. They learn so much from watching it: world geography, traditional music from all continents, various customs and how music can be a common theme for all of us to live together. Our grandkids become totally involved in this great presentation.

Rating: 5.0 and helpfulness: 1/1
<http://www.amazon.com/dp/B0001GH7V8>

Tone & Learn

This DVD is wonderful if you would like to learn yoga and gain strength/toning. The instructor is clear in her instructions and I was able to follow along. After using it awhile I found the poses were falling to place and felt myself progressing. A very good choice if you want to learn from a fabulous instructor and I am very happy with this DVD.

Rating: 5.0 and helpfulness: 0/0
<http://www.amazon.com/dp/B002VL2PTU>

MICHAEL JACKSON

Oh my goodness, Michael was going to bring London down to their knees. I am so sorry he did not do even one show and the complete concert was not filmed. I miss him so much, and I still cry but I know he is out of pain and no one can hurt him again. Get this DVD and see a master at work. May he rest in peace.

Rating: 5.0 and helpfulness: 0/0
<http://www.amazon.com/dp/B0002I82XW>

Glorious, Wondrous, the Greatest Rudolph of Them All!

Rudolph, the Red-Nosed Reindeer is indeed the most marvelous Christmas film ever! It is one of many superb animated films written by a great man named Romeo Muller. Rudolph is doubtless his best-loved work. I remember watching the film as a child, delighted and enraptured by the story that it told. Do you remember? Sam the Snowman as narrator (voiced and sung in the performance of a lifetime by Burl Ives); Rudolph the misfit and rejected reindeer; Rudolph's friend, Hermey the Elf, also a misfit because he doesn't like to make toys but wants to be a dentist; Yukon Cornelius, the prospector who befriends them both; the Abominable Snowmonster of the North (whom Yukon, in prospector slang, invariably calls a "bumble"); the Island of Misfit Toys, and its winged lion ruler, King Moonracer; and many other characters and story threads. The presentation, interweaving, and resolution of all of these elements is positively Shakespearean in deftness, wit, poetic beauty and brevity of expression, depth, pathos, joy, moral instruction, and the sense that all ends as indeed it should. Grounded in a deep and sparkling love of all creation, these qualities characterize all of Romeo Muller's films. If you will see it devotedly--indeed, if you will see any and all of Romeo's films--you will become a finer, more compassionate, and a more truly loving human being.

Rating: 5.0 and helpfulness: 0/0
<http://www.amazon.com/dp/B001PHA6PQ>

Unusual But Great DVD

We originally rented this dvd from the library and my 4 yr. old liked it so much, that when it was time to return it, we decided to buy one. The entire process was quick and easy and the quality was great. Very happy with the whole

experience.

Rating: 2.0 and helpfulness: 0/1

<http://www.amazon.com/dp/B000AE8L2U>

Terrible...just terrible

Well, I rented this flick just to see what it is like. The acting was non-existent. CGI effect was laughable. Plot was thin and predictable. The characters were a joke. The only good thing about this was Glori Anne Gilbert who portrayed the porn star one of the crew constantly fantasized about... OK, make that two good things... LOL! :)

Rating: 4.0 and helpfulness: 2/2

<http://www.amazon.com/dp/B000IMM3XW>

X-ceptionally good

Wow!Now this is one heck of a sci-fi\action flick!Brian Singer sure proved to be a man of many skills.X-men is by far one of the best comic book-to movie-adaptations with only the original Batman and Batman Returns being on the same level.Why?First of all it actually has a script(no Batman & Robin bull),secondly it's a believable plot.Then we have the characters:all of them are superb!Singer obviously did his homework cause these characters mirror the comic book heroes in an amazing way,both with the use of their powers and their way of thinking and actions.Then we have the action scenes which are breathtaking and put movies like M:I2 into shame(it's up there with the Matrix,honest) and special effects the likes of which we haven't seen in a while(again since Matrix).Plus the way Singer "ends" the movie is really smart 'cause he doesn't really close any chapters ,just saying "i'll be back with a sequel".The way it finishes kind of reminds me of the original Star Wars ,as if it's part of a series:In fact,the movie itself is just like a comic book issue..there is always next month!The only criticism i have for X-men is that it is very short.It couldv'e been a lot longer and there for better.That is the only reason i gave it 4 instead of 5 stars.Make part 2 longer please!

Rating: 5.0 and helpfulness: 3/5

<http://www.amazon.com/dp/B0000DK4QJ>

Brilliant "Report"

What do you get when you team one of the world's most bankable actors with the greatest director alive today? In the case of Tom Cruise and Steven Spielberg, the result is "Minority Report", a thrilling and intriguing crime drama which will most certainly become a sci-fi classic. The year is 2054, and Washington, D.C. has been murder-free for six years, thanks to the implementaion of Pre-Crime, in which gifted psychics (or Pre-Cogs) are used to predict murders before they occur. The Pre-Crime police force, led by Chief John Anderton(Cruise), are able to arrest and "halo" the accused before the crime is committed. Anderton believes fully in the pre-crime system, despite the skepticism shown by federal

agent Danny Witwer(Colin Farrell), who has been assigned to find flaws in pre-crime. It doesn't take long for Anderton himself to be predicted of committing a murder, which leads him to believe that Witwer set him up in order to make pre-crime fail. Nonetheless, Anderton must go on the run to prove his innocence...or is he unknowingly on the path to murder? This was undoubtedly the best film I saw in 2002(and I saw quite a few). It's a huge rebound for Cruise, who was last seen in 2001's dismal "Vanilla Sky". I never particularly cared for most of his other performances, but he is remarkably good in this film. Oscar-worthy? Quite possibly. And Spielberg demonstrates why he is the greatest director of the last 30 years. Yes, he's had some misses("1941", "Always", "A.I.:Artificial Intelligence"), but as with all his great films, he can thrill you one moment, elicit a laugh the next, even scare the pants off you, but his best films are the ones you talk about afterwards. No, "Minority Report" is not the most interesting title of a film, but it does figure into the plot. What's most surprising is the film's rating-somehow, it received a PG-13, when it should have been rated R for violence, suggested drug use, language and adult subject matter. However, it does not change my rating, as the film is a marvelous blend of special effects, top-notch acting and a fascinating story, which is hard to come by in a lot of films these days. See it!

Rating: 5.0 and helpfulness: 0/2

<http://www.amazon.com/dp/B0000DK4QJ>

A great film!

Minority report is the best science fiction film. The story is never boring and surprises. The actors act very well ! It's one of the best movie I ever seen
!
So I highly recommended it !

Rating: 5.0 and helpfulness: 3/3

<http://www.amazon.com/dp/B0000DK4QJ>

Do You Have One?

Minority Report definitely ranks as one of the best future-theme movies of all time. Although not as groundbreaking as The Matrix or visually stunning as Blade Runner, it has its own pros.<p>In a future where future crimes are predicted using triplets or 'Pre Cogs' with the ability to foresee events yet to come, the Police Officer in charge and support of the device himself (Tom Cruise) is accused of a murder that he has yet to commit. Though he believes that he is innocent, the squad doesn't, so Cruise sets out on his journey to find the truth.<p>The storyline for the movie is interesting, very thought provoking. A lot of what is talked about sounds complicated at first, but listening closely will reveal that they are really talking about the Minority Report of Cruise. Listen even more closely and you'll understand the entire movie, otherwise you will be lost.<p>Good actors playing good characters, impressive special effects, quality action, and a deepening plot - Minority Report is definitely worth the price of admission.

Rating: 5.0 and helpfulness: 0/1
<http://www.amazon.com/dp/B0060IW2NA>
Perfect

It is not a sugar coating type of movie like you see from Hollywood.
It is a down to earth story about two real people that had their fate altered by one event.
No exaggerated traits in their characters and their emotions.
No over the top nothing.
Even though they had only crossed paths once, their lives were fused by the existence of the other. An unlikely strong bond between them was created with negative effects for both. It was a bond that had to be broken but not in a way you would expect from a movie but how it would happen in real life.
See it.

Rating: 3.0 and helpfulness: 3/4
<http://www.amazon.com/dp/B004SIP6BI>
Deceptive or Realistic???

It's interesting to see that there's so much activity for this film since it's been released for quite awhile. However, perhaps Depp's recent performances have drawn attention to it. After watching him in "Pirates of the Caribbean," I realized that he might be one of the best young actors in our time. I mean he can really act like different people...not just take type-cast rolls. And so I wanted to see his work here.<p>What I found was a blatant skewing of a storyline to engage people while communicating a specific message. While most intelligent books & movies now try to show how multi-dimensional everyone is...how difficult it is to simply cast someone as "good" or "bad"...this movie does the reverse. As characters are introduced, you are quickly able to assign them to one of 2 camps: good enlightened folks or bad religious uptight folks. (There's also one bad drunken guy who is taken in by the bad religious folks for their own agenda.)<p>What unfolds, then, is not a film about the transformation of human nature or the triumph of hero in a realistic set of circumstances. Rather, you have an "unlikely" heroine cast in a role where she's bound to succeed in proving how good she ultimately is. (She has all good people on her side, of course). This mysterious woman who works on Sunday & has an illegitimate daughter basically acts as the religious figure of the movie. She heals the sick, protects the vulnerable, brings together the lonely and even, yes, loves her enemies. The bad people are eventually either "converted" to her side or made to look absolutely ridiculous. They are degraded and dismissed.<p>Depp's character makes an interesting entrance towards the end of this movie. As usual, I think he played his role well. He plays the intelligent, intriguing, competent gypsy he's supposed to be. I like his character, but I wished there was more of him.<p>In the end, I had very mixed feelings about the film. I felt that the artsy people behind the film (not necessarily the author of the original book) were using this film to express their own internal emotions. It was like they poured themselves into this heroine and tried to make you understand how good & sincere they are - despite their unconventional lifestyles. <p>And it is engaging. I legitimately liked the heroine & hoped she would succeed. I felt like I had to use all my intellectual power (which is, admittedly, minimal) to pull back from the emotional draw of the movie in

order to analyze the realisticness of the plot. As I said at the beginning, my ultimate feeling is that this is a blatantly skewed plot used to communicate a message. And that's not necessarily a bad thing...perhaps it's just good to be aware of.

Rating: 5.0 and helpfulness: 0/0

<http://www.amazon.com/dp/B004SIP6BI>

Mmmmm Chocolate...

Loved this movie. It's cute and simple, yet has a lot of messages about life and people and religious dogmas. Total chick flick. My boyfriend didn't make it through.

Rating: 4.0 and helpfulness: 0/0

<http://www.amazon.com/dp/B00005LKLN>

Looks Great!!

Have seen this title in SD, but in HD looks soooo much better. If you have a chance pick it up!!!!

Rating: 5.0 and helpfulness: 0/0

<http://www.amazon.com/dp/B00005LKLD>

Classic

Great movie brings me back to my childhood days. A feel good heart warming story, that is also very funny.
The remake does not come close to this one, This movie is eye candy, and it has a very good cast.

Rating: 4.0 and helpfulness: 0/0

<http://www.amazon.com/dp/B0014ERK00>

Necessary Ending, A must See for Stargate Fans

This movie was a must for the series. The end of the series left you hanging on many open issue. This movie answers may questions and completes the series as they should have done in the TV series. If you followed the Stargate SG-1 series you need to watch this ending.

Rating: 4.0 and helpfulness: 5/5

<http://www.amazon.com/dp/B000067J2J>

Silver Screen buff

John Payne, Dan Duryea and Lizbeth Scott gave tense, very good performances in this color western. John Payne was especially good at showing what can happen to a person as he is slowly stripped of his self-respect, reputation and every friend in town, who desert him. Finally, he is down to nothing but self-preservation and survival. Lizbeth Scott is Payne's loyal love interest and Dan Duryea plays a convincing and clever villain. There are insinuations of 'McCarthyism' in the film, but they are too simplistic to be taken seriously.

The career of Senator Joseph (Tailgunner Joe) McCarthy had a lot more to it than could be talked about here.

I thought the color balance, sound and image clarity were very good for this DVD.

Rating: 5.0 and helpfulness: 9/11

<http://www.amazon.com/dp/B00005U14H>

It's 4am...and I still can't stop thinking about it.

Before watching Bully knowing Larry Clark had directed it gave me a taste for what it might be like. Based on a true story I was taken in by the whole thing instantly. The story was a great basis for an even greater movie. Larry Clark's choice of actors was perfect. Bijou Phillips as the care free, experimental Ali was just right. Nich Stahl as the Bully, played it with such truthfulness, that in certain scenes was disturbing to see. Brad Renfro as Marty - the initial victim of Bobby's violence. Plays it with a perfect sense of fear but of also the need to rebel against him and eventually stop running. <p>In typical Larry style the movie uses very unusual camera angles to make things as real to life as possible. Larry Clark also likes to shock and have an impact on his viewers - small elements like the unbearable camera spinning round a group of teens for well over 1 minute made me turn away dizzy, also using an unwanted shot of Bijou's crotch, it served no purpose.
The movie is very real, Larry hasn't left any stone unturned. High amounts of ...frolics between the teens and even the murder itself are as real and are played with as much truth as possible.
I watched the first 10 minutes and I had to turn it off - because I was brought face to face with a grim reality - that was done in such a believable way that I had to register what I had seen. It was quite shocking to see some of the actors as their characters. For example Michael Pitt went from innocent and loving Henry in Dawsons Creek to a guy who has lots ...[of relations] and would commit ...[crimes] As uncomfortable as the scenes were, the acting was at such a high level you'd never know they were phased by it. <p>Although Bully is initailly a grim story of ...[crimes], I think Bully is on some level a fun, entertaining teenage flick, its the violent elements in the story that change the whole feel to it.
Bully has fresh new talent and a kicking soundtrack. Larry Clark has always been interested in youth..., in this movie he has combined that and applied it to a gripping story.
Bully has been a real eye opener. Its been a week or so since I watched it and I still can't stop thinking about it. Walking away from a movie and forgetting about it is a wasted 2 hours - walking away and thinking about it weeks after and wanting to know more - well thats a job well done.

Rating: 5.0 and helpfulness: 4/4

<http://www.amazon.com/dp/B00005U14H>

Man, what a good freaking movie.

I was blown away by this movie. And I'm rarely blown away by movies. There was almost NOTHING wrong about it. <p>Well shot, great cinematography, great script, great music. But I feel like saying all that lessens the sheer power it had and the immense feeling I felt watching it. It said so much about high school and

kids and life and it was just so freaking honest and true. <p>Maybe the best thing, and most delighting, about this movie though is the acting. Literally everyone is just so freaking good. Everything was just so real, it made me so happy and excited and engaged.<p>I can't say enough good things about this movie. Go see it right now. I'm just upset that no one saw it in the theaters when it came out. I heard stuff like, "it was needlessly violent." It's not even that freaking violent. The violence is real and actually made me cringe and feel something.<p>Anyway. Go see this movie.<p>B+, A

Rating: 1.0 and helpfulness: 0/0

<http://www.amazon.com/dp/B000NDFHYS>

Please, God, let these STAY lost

At this time, I am on what you may call a National Lampoon investigation. You may have noticed the makers of classics like Animal House and Vacation are releasing 10 movies a year in hopes to get a hit. So I have been watching their latest films. When I reached Lost Reality, the idea of shows that could never air sounded funny enough.

 Oh, it did, my friends, but after watching this, I had many emotions, none good. Disgust, confusion, and...well, more confusion. Is this supposed to be funny? Are these real show ideas compiled by NL? On the box it says there are 'creators' for the film. Creators of the made-up pilots, creators of a show that found horrible show ideas, what? Who can I blame for this travesty?

 I know these shows are SUPPOSED to be horrible, horrible shows, but aren't they also supposed to make the viewer laugh? Watching someone drink a liquid that makes you barf to get cash and then lick it up is toe-curling whether it's real or not. It LOOKS real (they're filmed like actual reality shows), so are these twisted people doing this to real people? Take 'The Whore,' where three men are set up with a paid prostitute. Do these men know about this? Who's in on the joke? The reason shows like Punk'd, Candid Camera, or the Jamie Kennedy Experiment are (or in some cases were) funny is because we're in on the joke with Ashton Kutcher and the actors he's using, we know it's Jamie Kennedy, and so on. Also, the gags done are funny. In the excerpt in Lost Reality called 'Casting Couch,' we have a man hiring women for movies and such based on whether or not they will read topless. Are these women aware? Is this man subjecting real (admittedly gullible and stupid) women to this? I somehow managed a chuckle or two, only because I saw what kind of shows they were spoofing. Reality shows can be brain-numbingly cheesy, which 'He Said, She Said' explores until it goes overboard (and below overboard, and below, and below...). And just when you think it's over, it throws in another 'Money' sketch, where a man craps his pants for money, and the camera ZOOMS IN ON THE CRAP.

 All I can say is that National Lampoon has lost its touch whether this is real or not, because if it's fake, it's not funny at all, and if it's real, these men have no, repeat, no compassion for humanity.

Rating: 5.0 and helpfulness: 10/10

<http://www.amazon.com/dp/B008SAPAYA>

A Silent David and Goliath

Tol'able David is one of the finest silent films which I have seen. It seems more realistic than many silents. This is partly because of the location shooting, partly because the story is neither overly melodramatic nor comic. Barthelmess as the David character gives a wonderful performance while Torrance as the Goliath character is only just a little over the top in his villainy. He was perhaps more suited to the comic monster type of his later Captain Hook. The real revelation though, is Hulette. She is almost completely forgotten now, but clearly had great talent. Her performance is realistic and charming. She is quite simply adorable. The print on this DVD has been formed from separate sources. Thus it looks slightly different from scene to scene. There is some damage, but it rarely detracts. It is however, slightly distracting to see variations in the tinting. It is inconsistent to have one green scene when the rest of the movie uses greys and browns. One of the best features of this DVD is the music, which is a series of wonderful folksy tunes which fit in so well with the action that even David's harmonica playing is included. As an added bonus the DVD includes an interesting interview with director Henry King and some useful, extensive sleeve notes. Tol'able David is not as well known as many silent films, but really it ought to be considered as among the very best.

Rating: 2.0 and helpfulness: 0/0

<http://www.amazon.com/dp/B001UHXL5S>

The New Beginning 2012

Its something about this film that bother me thats because im a film makers im not here to judge those involved in the making of this film i would say if redone today that the result will be different as fur as directing and acting is involved in this production i would like to redone this film again if possible what was lacking was the story lines that didnt grab the attention of the viewers and professional camera equipment ...! Sign GM

Rating: 5.0 and helpfulness: 0/0

<http://www.amazon.com/dp/B005B5PD92>

Review of Armadillo

Overall excellent production. Presented a meaningful contrast between the various expectations of the participants and their factual experiences. First there was elation, then letdown, and finally acceptance. Some of the participants seemed to mature right before our eyes, thus proving that war is truly Hell.

Rating: 5.0 and helpfulness: 0/0

<http://www.amazon.com/dp/B003SHYYM2>

Good Family Movie

Holes, based on the best-selling and award-winning children's book by Louis Sachar, is a clean, wholesome adventure that is sure to please every movie-lover in your family.<p>Fans of the book will appreciate the effort made to keep the movie true to the book. They did a great job with this, perhaps because the

author is also the screenwriter.<p>The story is easy to follow even for the youngest in your family (my four year old daughter loved it). The story is simple, yet exciting and captivating enough for the older crowd.<p>I highly recommend this movie to anyone looking to add a good family picture to your movie collection.

Rating: 1.0 and helpfulness: 3/8

<http://www.amazon.com/dp/B00005JMZK>

Paul Anderson!!!!!!!!!!!!!!!!!!!!

I haven't seen this film and I have no intention of it. Yes, I am a huge Alien fan, and yes, I am a huge Predator fan. I am also a fan of the Alien vs Predator comics, and to a lesser extent, the video games. I have been awaiting the release of this movie for as long as any other die-hard fan out there, however something completely ruined it for me that can never be put right. Those morons in hollywood ran a few names of successful directors in hollywood at the time, and out of what must have been a very short list, some no brained prat asked Paul Anderson if he wanted to direct it.

Paul Anderson...

How can you give this film, of all films, to Paul Anderson. He simply can't direct. Anyone that can take a format as simple as an empty house with a bunch of zombies, and make Resident evil out of it (One of the worst films of all time) obviously has no sense of how to make a film whatsoever. He also wrote the only review of the film that appears on any of the posters. Doesn't that say something? That nobody would give it a good review so he had to do it himself.

Oh well. It good have been great this film, but that's what happens when money is the key issue.

Doesn't matter anyway, I'm sure that all those 8 year old kids that can legally watch it on their own will enjoy it.

Rating: 2.0 and helpfulness: 1/11

<http://www.amazon.com/dp/B003DBEX6K>

Vegetarian Vampires Are Human Too

The "Twilight" books don't hold up very well if you compare them to "Salem's Lot" by Stephen King or "They Thirst" by Robert McCammon. I don't want to insult anyone by saying they are pretty much total garbage, but they are. Teenage girls will be squirming in their seats just to see Jacob's Abs of Steel when he takes his shirt off. Poor Bella can't decide. Who does she love more, Edward an Jacob? Oh, the agony!

Rating: 5.0 and helpfulness: 0/0

<http://www.amazon.com/dp/B008U05U9E>

Great

This is a great movie. It has a good plot. Its definitely worth seeing if you have not. Jason Statham is a good actor for the the new generation.

Rating: 1.0 and helpfulness: 3/5
<http://www.amazon.com/dp/B0001NBNF2>
absolutly awful

This movie tries so hard to be cool, but it makes the film even worse. The action is a copy from the matrix and the plot is so stupid and people thought gigli was bad, this is way worse. Doesn't make any sense.

Rating: 5.0 and helpfulness: 0/0
<http://www.amazon.com/dp/B0006HBZXS>
Very Good!! Don't think he(Hopkins) ever does anything of inferior quality and this doesn't disappoint.
Try it if you like getting away from it all... Good combination of wilderness and intrigue. Good cast and doesn't having Elle around.

Rating: 5.0 and helpfulness: 0/0
<http://www.amazon.com/dp/B004HRDMHK>
Your second piece is even better!!!
What can I say...this movie is great!! A year has passed since prom and the summer has arrived...this means fun fun fun!
The same cast come together for the trip of a lifetime at the lake..cue outrageous goings on and non stop laughter. Although they have all moved on in life, they are all the same sexually frustrated people who left high school a year ago. Stifler has not changed one bit...he's even more ready to spread himself among young girls. Jim is back to his normal self...making a fool of himself with unusual scenerios which often lead to total humiliation..but this summer things start to change for the better. Oz remains in love with Heather and battles a whole summer without her...cue interesting phone conversations!! And Finch...still obsessed with the one and only Stiflers Mom. Lots of funnny moments and even more outrageous goings on. Many people thought AP2 would not satisfy but hey I think its great!
Keep going back for seconds..you know you can't resist!

Rating: 5.0 and helpfulness: 0/7
<http://www.amazon.com/dp/B00015HX9A>
response to Mari
I have to say that I fully agree with Mari on the similarity between Sarah Jessica Parker and the poor child sitting inside the dilapidated chapel. When I first saw "Brother Sun, Sister Moon" this person immediately caught my attention as being someone I had seen before. Needless to say, Sarah Jessica Parker was the name that came to mind. Although I know for sure that it isn't the same person, I can see where the confusion would be...

Rating: 5.0 and helpfulness: 0/0
<http://www.amazon.com/dp/B000J0XJC2>
Great!

Got this for my mom, but it's on TV every Easter week. Did a great job with this movie for it's time.

Rating: 5.0 and helpfulness: 2/4

<http://www.amazon.com/dp/B000J0XJC2>

DeMille made the Bible come alive

Before I ever was ordained as a minister, I saw this movie on the movie screen three times, and have seen it many times since I was ordained. (Thus, I have seen it a countless number of times between the ages of twelve and my present age of sixty-four.) I'm glad I have the dvd of it now. While the scene of the meeting of Moses and God at the burning bush,... and as well the scene of the parting of the Red Sea...stick out in my mind, all through the movie I felt a euphoric "you-are-there" feeling of the story which took me back to 1500(?) B.C. Though liberties were taken with the drama, it did follow very closely the story line in the Bible.
Did any movie, barring none, ever have a more perfect cast? Charlton Heston was arguably the best for the role of Moses, the strong, awe-inspiring, sometimes hot-headed leader of the children of Israel. The Pharoah was just as mean as Yul Brynner, I dare say. Anne Baxter, (whom I always considered a "put-on") was convincing as the tempting Nefretiri. Yvonne DeCarlo was adept as the physically attractive, sweet, and understanding Sephora. (In the Bible, the name is actually "Zipporah".) Martha Scott was molded to play Moses' mother Jochebed. John Derek definitely was the strong type which fit Joshua. The seasoned actor Sir Cedric Hardwicke acted very well the older Pharoah Sethi. Vincent Price was the "perfectly" mean and cold-hearted task master Bacca. Edward G. Robinson portrayed well the traitor Dathan. Debra Paget was adorable. Nina Foch, who assumed roles of both heartless and loving women, did a very good turn as Pharoah's daughter, Moses' adopted mother who was so loving. Again, the entire cast was par excellence.
Naturally and needless to say, the color and the "trick photography" of so many of the scenes also made the movie so realistic and enhancing.
All the way through the story...from Egypt, to Midian, to Mount Sinai, to Egypt again, to the Sinai desert and to Mount Sinai again, and then to the Promised Land (which Moses was forbidden to enter) the story of Moses and the exodus was presented so well. For Cecil B. DeMille...who wrote, produced, and directed the film...this was his magnus opus. (Winston Churchill might have said that this was his "finest hour".) Seriously, to say that it was a magnanimous movie is understating the case. A superior drama.

Rating: 5.0 and helpfulness: 1/6

<http://www.amazon.com/dp/B000J0XJC2>

blu-ray Ten Commandments

As good as the 1957 version is. This version takes it to a much much higher level...
Sound quality was much better than the original but not today's standards

Rating: 5.0 and helpfulness: 2/3

<http://www.amazon.com/dp/B00005V9IK>

Your Typical Action / American Film - But It's Great!

Firstly, you're amazed by the alien ship that appears, curious about the contingency plan, and then completely blown away by the explosions and special effects. This is American patriotism at its very best, yet you forget all of that with the amount of action, comedy and typical plotlines. Ok, so the acting wasn't THAT great, but you can watch this film a hundred times and never get bored. There one liners always crack you up, and once again, America is always the one that saves the day! <p>A simple plot - aliens threaten the world's (more so America's) existence and someone (America) has to do something about it. With a cast led by Bill Pullman, Jeff Goldblum and (at the time) an aspiring Will Smith, that were all very good to watch, not forgetting the rather nice looking Vivica A. Fox as Smith's fianc? I can't really put a flaw in this film since it was great fun to see. God bless America! <p>Done in true US Military style, 'Independence Day' is an entertaining, patriotic watch - one big dose of eye candy which can be seen again and again.

Rating: 4.0 and helpfulness: 2/2

<http://www.amazon.com/dp/B00005V9IJ>

Take Nothing for Granted!

If you're a fan of Tom Hanks, than this movie is a must-see! A story of a man who loses everything important to him in his life, because on one fateful plane trip he is left stranded on a remote desert island...for FOUR YEARS!

Much of the movie early on provides foreshadowing for events to arise later. There is the love story intertwined within Hanks struggle to survive (albeit they are a married couple). However, like the rest of the film, even that doesn't turn out the way you'd expect!

My only gripe about the film is, I wish they would have shown a bit more of Hanks' on the island. He was there for FOUR years and the way it was filmed, it jumps from the beginning of his new life, to shortly before he is finally saved. I would have liked to have seen a bit more of the time in between. I can understand however because it was a lengthy movie as is and that would have only made it longer.

Check this out if you can. After seeing this movie, it'll definetly make you appreciate things a bit more and maybe help you not take so much of your life for granted.

Rating: 5.0 and helpfulness: 0/0

<http://www.amazon.com/dp/B00015HX90>

Great!

Got this for my mom, but it's on TV every Easter week. Did a great job with this movie for it's time.

Rating: 5.0 and helpfulness: 2/4

<http://www.amazon.com/dp/B00015HX90>

DeMille made the Bible come alive

Before I ever was ordained as a minister, I saw this movie on the movie screen three times, and have seen it many times since I was ordained. (Thus, I have seen it a countless number of times between the ages of twelve and my present age of sixty-four.) I'm glad I have the dvd of it now. While the scene of the meeting of Moses and God at the burning bush,... and as well the scene of the parting of the Red Sea...stick out in my mind, all through the movie I felt a euphoric "you-are-there" feeling of the story which took me back to 1500(?) B.C. Though liberties were taken with the drama, it did follow very closely the story line in the Bible.
Did any movie, barring none, ever have a more perfect cast? Charlton Heston was arguably the best for the role of Moses, the strong, awe-inspiring, sometimes hot-headed leader of the children of Israel. The Pharoah was just as mean as Yul Brynner, I dare say. Anne Baxter, (whom I always considered a "put-on") was convincing as the tempting Nefretiri. Yvonne DeCarlo was adept as the physically attractive, sweet, and understanding Sephora. (In the Bible, the name is actually "Zipporah".) Martha Scott was molded to play Moses' mother Jochebed. John Derek definitely was the strong type which fit Joshua. The seasoned actor Sir Cedric Hardwicke acted very well the older Pharoah Sethi. Vincent Price was the "perfectly" mean and cold-hearted task master Bacca. Edward G. Robinson portrayed well the traitor Dathan. Debra Paget was adorable. Nina Foch, who assumed roles of both heartless and loving women, did a very good turn as Pharoah's daughter, Moses' adopted mother who was so loving. Again, the entire cast was par excellence.
Naturally and needless to say, the color and the "trick photography" of so many of the scenes also made the movie so realistic and enhancing.
All the way through the story...from Egypt, to Midian, to Mount Sinai, to Egypt again, to the Sinai desert and to Mount Sinai again, and then to the Promised Land (which Moses was forbidden to enter) the story of Moses and the exodus was presented so well. For Cecil B. DeMille...who wrote, produced, and directed the film...this was his magnus opus. (Winston Churchill might have said that this was his "finest hour".) Seriously, to say that it was a magnanimous movie is understating the case. A superior drama.

Rating: 5.0 and helpfulness: 1/6

<http://www.amazon.com/dp/B00015HX90>

blu-ray Ten Commandments

As good as the 1957 version is. This version takes it to a much much higher level...
Sound quality was much better than the original but not today's standards

Rating: 4.0 and helpfulness: 0/0

<http://www.amazon.com/dp/B005LAJ22Q>

A different take for the Walking Dead

Walking Dead has been hugely popular in the comic scene for awhile now, yet the introduction of this AMC show skyrocketed the whole franchise into fame and notability among not just the 'nerdy' markets yet the TV market itself. This

surge of popularity has given the franchise an amazing amount of room to branch out to video games, novels, collectibles, etc. So the second season of the TV show was bound to have a lavish collectors edition, and while it isn't a letdown, there are some annoying drawbacks.

For start, the "head-case" is pretty cool. It does come from a pretty iconic scene exclusive to the second season of the Walking Dead. The blu-ray discs fit perfectly inside, and the opening/closing mechanism seems pretty solid, yet I keep them with the other blu-ray discs because I see no reason to hide them inside the head. The detail of head seems similar to a high quality Halloween mask up-close, yet looks fantastic from a distance away, and I also love the big title-words underneath the head. Overall, it looks really awesome on a shelf, I have mine on my shelf dedicated to Walking Dead where it holds all my books, issues, and blu-rays of the franchise.

The season itself is pretty awesome, I've found that most readers of the book wish it stuck closer to the book and went through these events much faster. (For comparison, the first two season of this show only cover the first twelve issues of the book, and each issue is very short.) Yet I find the second season to be a more dramatic, realistic, and character driven version of the book. There's more time for each character, and more pacing for each event. There's still certainly some issues with the show, most notably two of the characters are jaw-droppingly more stupid in the show, and some other characters are seemingly not in the show.

The video and audio quality is terrific. Due to the nature of the show, there's a very gritty look with plenty of film grain, yet it all looks great. There's a general lack of a soundtrack, so there's plenty of time to notice all the immersive background sounds, most notably a very subtle effect of a horde of zombies moaning into a collective disturbing feeling that is used numerous times. While the more intense sounds of gunshots and yelling all stay crisp and clear.

This release is spread along four blu-rays, so it's obvious there's a very decent amount of extras here. Yet, the first season had at least one audio commentary on each episode, while this one only has audio commentary on five of the thirteen episodes (and the 20 minute collection of the webisodes). The rest of the special features are pretty awesome, a huge amount of behind the scene and interview featurettes which each clock in around seven minutes. The best part is probably the 30 minutes of deleted scenes.

Now while this release is pretty awesome, there is a looming threat of there being another release of the second season in order to get another \$40-\$50 out of the diehard fans. This was done with the first season, and it could very well be done with this one with the rest of the audio commentaries. Yet even if that does happen, one could still be happy with this release just for the zombie head-case.

Rating: 5.0 and helpfulness: 0/1

<http://www.amazon.com/dp/B005LAJ22Q>

Best TV Show EVER!

This show is gripping, intense and shows us what an amazing story line is!

Brilliant show. Andrew Lincoln is at his best and the rest of the cast are also phenomenal. Producers of the show have done brilliantly.

Rating: 5.0 and helpfulness: 9/13
<http://www.amazon.com/dp/0767805534>

Freedom

It is as simple as this. There's nothing, really, to "get". This movie is about freedom and peace. It's about how the tyranny of the "Status Quo" stifles the spirit of man, and with bleak outcome, illustrates how this wreckless hatred of difference, and unacceptance of our fellows, ends in destruction of peace. Those with baseless pride, always quick to judge and condemn, are the real villains, despite the protagonists' lawlessness, they bring no harm to anyone, which is in stark contrast to the majority of those they encounter. <p>I keep reading reviews on how many "sided with the rednecks". All I can say is that if you find yourself doing just that, then you really must delve further into the nature of freedom. *REAL* freedom. <p>That is the essence of this film. It is the one word of description to label it and define it. The protagonists' are on a quest to discover what it is to be free, whilst indulging in it. The realisation of the farmer's existence is an example of where they see how freedom is manifest in different forms. Despite the choice of the expression, the farmer and the (anti)heroes share a similar viewpoint on life. <p>This may very well be the most patriotic film ever made. It is a shame that so many can't look past the surface to see that fact.

Rating: 4.0 and helpfulness: 0/0
<http://www.amazon.com/dp/B00004CZRQ>

The most terrifying film of all time

I have no doubt that this is the scariest movie ever made. No stone was left unturned when William Freidken and team wanted to make this a film that would burn itself into you memory. This is a movie that will effect you emotionally in a number a horrifying ways. The DVD looks and sounds great. Proof that the sound of a film can and will have just as much impact on you as the visuals will. The bonus material is first rate. Get this DVD. Turn down the lights, put the kids to bed, turn up the home theatre and get ready to be terrified beyond belief !

Rating: 5.0 and helpfulness: 0/0
<http://www.amazon.com/dp/B00005LL26>

Classic

Great movie brings me back to my childhood days. A feel good heart warming story, that is also very funny.
The remake does not come close to this one, This movie is eye candy, and it has a very good cast.

Rating: 5.0 and helpfulness: 0/0
<http://www.amazon.com/dp/5553600715>

Great Movie

This movie was everything and more of what I had expected. There aren't too

many movies that can make the hair stand up on the back of my neck, but there was one scene which did exactly that.
I don't think anyone will be disappointed.
This was the movie, I think Nicole Kidman should have been nominated for(academy award). Great performance!

Rating: 4.0 and helpfulness: 1/2
<http://www.amazon.com/dp/5553600715>

Les autres

The story is good, and it's like The sixth sence, but not as good.
the actor act well. the story is complicated but easy to understand.
You never get bored and sometimes it's amaze.
So I suggest you to buy it.

Rating: 4.0 and helpfulness: 1/2
<http://www.amazon.com/dp/5553600715>

Back to Classic

I purchased the Asian DVD not long ago although I had never seen the film in theaters. I had heard enough about it, but didn't want to know too much to ruin it. I was very pleased to find that the movie uses very basic techniques to their tried-and-true results of scaring the pants off you! It seems in these days of hi-tech effects the filmmakers have to explode in the face of the audience to get a thrill. Using suspense, excellent acting, and a good script this classic ghost story will truly satisfy. AND, finally a film for all ages that does so.

Rating: 5.0 and helpfulness: 1/1
<http://www.amazon.com/dp/B000ZLFALI>

Perfect musical of the year!!

Normally I don't write review but I must say "Across the Universe" makes me smile, makes me laugh with joy and makes me cry. It is a kind of film that you cannot help but watch it again and again. Excellent cast, great direction and exceptional music arrangement. Highest recommendation in every way!

Rating: 5.0 and helpfulness: 2/4
<http://www.amazon.com/dp/6300182150>

The Negroes took our dates!

All college movies begin and end with a little film called National Lampoon's Animal House. A foray into movies for the first time for National Lampoon, having succeeded on radio and as a magazine, this raucous classic has everything you'd want in a comedy: memorable characters & quotes, partying, a great soundtrack, and a group of unknowns that became stars overnight.

 The basic plot: freshmen Kent Dorfman (Stephen Furst) and Larry Kroger (Tom Hulce) are rejected from the preppy Omega fraternity and head to the Delta house, where Kent's older brother Fred went. Delta turns out to be the party frat, aka 'animal house,' at Faber College, and Dean Wormer (John Vernon) wants them shut

down and puts them on double secret probation. But Delta won't die, so one toga party, road trip, horse killing, food fight, and parade destruction later, the Deltas beat the rules.

 Saying Animal House always brings to mind John Belushi as Bluto Blutarsky, the 7-year Delta member whose grade point average is 0.0 and who gives fine advice like, "Start drinking heavily." Bluto is the best character hands down, and his speech to get the Deltas back up when they're down is classic, yelling, "Was it over when the Germans bombed Pearl Harbor? Hell no!"

 So if you haven't see Animal House, you basically haven't lived. See it now, then watch it again. GATOR!

Rating: 5.0 and helpfulness: 1/1

<http://www.amazon.com/dp/B000ZLFALS>

Perfect musical of the year!!

Normally I don't write review but I must say "Across the Universe" makes me smile, makes me laugh with joy and makes me cry. It is a kind of film that you cannot help but watch it again and again. Excellent cast, great direction and exceptional music arrangement. Highest recommendation in every way!

Rating: 4.0 and helpfulness: 1/2

<http://www.amazon.com/dp/1557832854>

Waiting for Godot, Maxnotes

I had the joy of seeing Waiting for Godot performed on Broadway last year and wanted a copy of the script. I found out when these notes arrived that it was not the script, but I decided to read it anyway and I am very glad I did. For anyone who has enjoyed this work of Beckett's these notes add a dimension to the experience that makes it richer.

Rating: 4.0 and helpfulness: 0/0

<http://www.amazon.com/dp/B000068TWC>

Two solid Arnold movies

While neither of these movies compare to classics such as "Total Recall" or "Terminator 2: Judgement Day," they are still solid action films.

"Eraser" features Arnold as a CIA agent who has the job of erasing witnesses backgrounds in order to protect them. When Vanessa Williams is targeted by the villians, he must protect her in order to save the country. This is a good film with some great action scenes. This also features some of Arnold's better acting.

"Collateral Damage" is another entertaining film. Arnold stars as a fireman who goes to Columbia seeking revenge. The film was originally planned for an October 2001 release. However, the theme of terrorism took a different tone after September 11, 2001. Wisely the film was postponed for a while. I found Arnold's acting to be a little off in this film, especially toward the beginning where he has to communicate grief and loss. Once he gets to Columbia, the film picks up quite a bit and is a very good action film.

The DVD set is a flipper. "Collateral Damage" has a nice collection of special features. Both films are presented in widescreen.<br

`</code>
When I originally saw "Eraser", I felt that maybe it was time for Arnold to retire (I like it more now than I did then). He continued to star in films, but the returns at the box office were diminishing. "Collateral Damage" faired poorly at the box office, although I think that had as much to do with bad timing than a loss of interest in Arnold's films. However, in the latter, the Austrian Oak seems a little tired of the whole genre. After the success of "Terminator 3" he would run for governor. With his term coming to a close, perhaps he will return to making movies.`

Rating: 4.0 and helpfulness: 6/9

<http://www.amazon.com/dp/B00008XFAM>

subtle point - not a full review *CONTAINS SPOILERS*

I both read the book and saw the movie years ago when it was first released. My recollection is that while Jesus does "marry and/or consummate" with Mary Magdeline as many of the reviewers before me have mentioned, she immediately dies, and he then marries Mary the sister of Lazarus, with whom he raises a family.

Regardless, I found the duality of man/God as presented in both the book and the movie to be incredibly moving. In fact the "temptation" placed before Jesus in this story seems far more powerful to me than even those placed before him in the books of the New Testament.

All in all, I loved the book and the soundtrack, and liked the movie.

Rating: 3.0 and helpfulness: 1/1

<http://www.amazon.com/dp/B000XJSL9U>

Don't worry, America. Bruce is back.

Bruce Willis brings his A game to this movie and it's one of the summer's better shows. Nobody can take it and dish it out better than Bruce Willis. Of the 4 movies in the series, I'd rate this one a strong #2. Yes, it's over the top non-stop action, but no one goes to see these shows expecting another Masterpiece Theater production of a Jane Austin novel. The stunts are the thing and this movie abounds in them. Bruce and his computer geek boy wonder work surprisingly well together. The only thing that keeps the movie from all time action film classic status is a fairly lackluster performance from Tim Olyphant, suprisingly so for fans of Deadwood. To make John's heroics truely meaningful he must have the proper foil, a worthy enemy. Alas, Tim is no Alan Rickman, who really made the first film so memorable. Neither Tim nor the script writers did anything unique or especially interesting with the character. Otherwise, I would have given this at least 4 in the genre.

Rating: 1.0 and helpfulness: 11/25

<http://www.amazon.com/dp/B000XJSL9U>

He's an easy guy to like, but a tough man to kill...

Well, he at least should be an easy guy to like, and has been for 20 years. In this however, the wonderful character of John McClain failed to make an appearance. First off, no swearing. Well, I don't care if a movie has swearing

or not, but this is Die Hard, and that's the Die Hard character. He's a guy's guy, he's brutal, he breaks peoples necks, tells them he's gonna cook and eat them after they're dead, refers to people as 'f**k face' a lot and has one of the most well known movie lines of all time, which incidentally, includes the word f**k. So, why did they have to make it without swearing? Seriously, from an artistic point of view, or simply in respect of the character and the films that came before it, it should have had swearing. Bruce Willis isn't too old for the part, but the script was lousy and played off like he was. They just made him a dumb computer illiterate who everyone thought was stupid. Stuff like covering a web cam and whispering isn't funny, it's just ruining the integrity of the character. He felt more like a super hero than a real guy, never once had an expression of fear or immediacy on his face, and was always happy to go to the next location when John McClain's role in every movie is that he always wants out.

First major problem with the film was the script. There was no hook, no one thing for him to get his teeth into. The movie wasn't claustrophobic at all and there was no time constraint. The result was that it took place over a fairly long period of time for a Die Hard film, rather than one evening to night or during the day. Every film is done in one go, in one bad situation. Die Hard with a Vengeance has the line when Samuel L' Jackson says 'You're about to have a very bad day' and us Die Hard fans were thinking, 'Alright, here we go again'.

Well, not this time.

The bad guy in the film was probably the worst bad guy I've ever seen, and that goes for all movies. He was so weedy and wimpy. There is no way in a million years that he would have kept McClain's daughter alive, or handed her the phone at certain points. She didn't need to be there, and he would have killed her.

God bless Kevin Smith, but he didn't belong in this film.

Why is it, that at the start of the film, the bad guys were sending viruses to kids houses who were playing Gears of War (Nice product placement there). But seriously, what's the point? John McClain knocks on the guys door before the pc explodes, the kid answers it because he doesn't know anyone wants him dead, so why can't the terrorist guys just knock on these kids doors, pop a gun between the crack of it, shoot them dead and walk out? Might attract less attention than a massive explosion!! But no, we have to settle for the PG rating virus explosions, because people can't die properly in films anymore.

Why was the film so shiny? Why were there so many unnecessary cuts? It seems like all movies today are made for people with ADD. Where's the pacing, the build up, the silence before the storm?

Why was the dialog so awful? Watch the scene in the limo in number 1 with Argyle and how great the banter between them is. Then watch the awful, awful, AWFUL dialog he has in the car in this one, when Justin Long says something along the lines of 'You watch the news!!! Oh my God, you watch the news!! What's wrong with you, do you know how much they lie to you on the news!!!! The news! My God, you watch the news!!! And you listen to popular music!!!!' ...

Jesus.

Somebody actually got paid to write that? I was sitting there thinking 'I'd rather watch the news than this lousy excuse for a Die Hard film'.

Why, of all things, was the Die Hard title screen different to how it's been in every movie so far? Die hard always slams in from the left and right of the screen, but not in this, they just said, 'no, screw that, forget the old films, no one cares'. Well, how would

you like it if someone changed the opening STAR WARS title, and just blipped it on instead?

This is easily the worst of the four, and Die Hard 2, despite me loving it, isn't a great film and shouldn't have been too hard to top or at least equal. All those that say Vengeance was weak, I suggest you watch it again. The script is great, and it's the only film directed by John McTeirnan aside from the first and it shows through.

John McClaine is a great character. The Die Hard films are fantastic action films. His lines stick forever, for example;

'Come out to the coast, we'll get together, have a few laughs'
'Thanks for the advice'
'Welcome to the party pal'
'No f*****g s**t lady, does it like I'm ordering a pizza!'
'If you're not part of the situation then you're part of the problem, quit being part of the f*****g problem and put the other guy back on'
'Ehh, sorry Hans, wrong guess, would you like to go for double jeopardy where the stakes could really change?'
'What's this a tag team?'
'For once I'd just like a nice quiet Christmas ... but here I am, climbing around in f*****g tin cans'
'How can the same s**t happen to the same guy twice'
'Well, now that you mention it, I have been feeling a burning sensation between my toes'
'He'll feel better when he looks in the back seat'
'Laundry day'
'Anyone catch the lottery last night ... see, I got the ticket right here'
'Say hello to your brother'
'Yippee Ki ya Mother F****r'

These are only a few, but you find me anything in Live Free or Die Hard that even comes close to these lines or scenes. All we get is him doing his McClain laugh since the script writers are practically brain dead and waste every opportunity for a good line. People have told me that a great line in this one is 'I'm gonna go kill this guy and get my daughter back, or get my daughter and kill this guy.'

Do I really have to comment on that? It's like he can't speak English. He just said the same thing twice, and neither time was it even witty!

The problem nowadays, is that Hollywood sell their films based on the action and the effects. It was nice to see real stunts in this film, but so what? He wasn't jumping off the top of a building as it explodes behind him, and what was real in the film was so badly edited that the stunts lost their impact. Way back when, there was a nice balance. If they wanted to make an action film, there was no CG, so they had to do the stunts. If they were going to take the time to get them right, then they were going to make damn sure the script was good so the stunts were worth doing. Sadly, this no longer seems to be the case, and Live Free or Die Hard was just another action film full of plot holes and no substance.

Oh, and that Bowman guy sucked.

Rating: 5.0 and helpfulness: 1/1

<http://www.amazon.com/dp/B006QF0SDM>

Brainy and Enlightening

This movie is revealing, hilarious, smart, a little bit madcap and touching all at once. I loved it. It opens up a world I don't know much about and presents characters that I love knowing about. The road trip storyline is almost a romp at times, but the film has emotional depth (the roles are really well-crafted), cultural insights (Mexican immigration), documentary qualities (the perils of border-crossing) and some killer visuals, too. So it's an all-around awesome

movie.

Rating: 5.0 and helpfulness: 0/1

<http://www.amazon.com/dp/6301976029>

Taras Bulba

A very good motion picture starring Yul Brynner and Tony Curtis in an classic movie set in the Russian era.

Rating: 5.0 and helpfulness: 0/1

<http://www.amazon.com/dp/B000AOGXH4>

Clint "CHEYENNE" Walker-Adult Western.

If you like Adult Westerns, then CHEYENNE is for You.
If you like Dramatic Westerns, then CHEYENNE is for You.
If you like Action -Charged Westerns, then CHEYENNE is for You.
If you like your Hero to be Strong and Handsome, then CHEYENNE is for you.
If you like to be Entertained, then CHEYENNE is for you.
 In other words:-
 Get It.
 Got It.
 Good!
And yes, I am a Danny Kaye fan as well, courtesy from The Court Jester.

Rating: 5.0 and helpfulness: 2/2

<http://www.amazon.com/dp/B0083SI986>

Well done!

Gran Torino was, in this person's opinion, Clint Eastwood's finest movie. I won't divulge any of the story, but it was very true to life for the older generation; a way of life that no longer will exist for young people. All I can say is "well done"!

Rating: 5.0 and helpfulness: 3/3

<http://www.amazon.com/dp/B0083SI986>

Very subtle and touching movie about the effects of war...

Several things: this movie is not what I expected. I had some vague idea there was fighting involved, and perhaps some racing around, and figured there would be a big "showdown" ~ after all, it's Clint Eastwood.

Also, I once had a huge fight with my ex, who wanted me to see "Saving Private Ryan". His argument: "War is an important topic and you need to see this movie." Mine: "I know it's an important topic, and I don't *need* to see a movie to know it."

So, having said that, again, not what I expected. Although I am can be tender-hearted, I rarely cry at movies. I did at this one. And I think people *should* see this movie to learn what happens when the war is over.

Clint's character is an old, racist, white man, who, after fighting in Korea, spent his life working for Ford. He has an old Ford truck, and a 1972 Gran Torino, his pride and joy, that incidentally he helped build. His wife has passed on, his kids hate him, he grandchildren only care about what he might leave them, he practically has no friends left, and his neighborhood is now filled with Hmong. Foreigners. Who he hates, wishing they would just leave the

neighborhood - or at least keep their yards up.

His neighbors hate him, and wish *he* would leave the neighborhood.

Stalemate, until the young boy next door tries to steal his Gran Torino as his initiation into a gang, and he catches the boy doing it.

I won't go too much into the details, but of course, bad things happen then. Good things as well. And we learn, in a subtle way, how that war affected him. And his family and children. And his foreign neighbors, and their children. And what "sacrifice" means. I highly, highly recommend this movie. It will touch your heart.

```
[12]: # Get best and worst rated movies
reviews_by_movie = reviews.map(lambda r: ((r['product_id'],), r['score']))
avg
    = reviews_by_movie.mapValues(lambda x: (x, 1)) \
        .reduceByKey(lambda x, y: (x[0] + y[0], x[1]
    ↪+ y[1]))

avg = avg.filter(lambda x: x[1][1] > 20 )
avg = avg.map(lambda x: ((x[1][0]/x[1][1],), x[0])) \
    .sortByKey(ascending=True)
```

```
[13]: for movie in avg.take(10):
    print ("http://www.amazon.com/dp/" + movie[1][0] + "    Rating: " +
    ↪str(movie[0][0]))
```

http://www.amazon.com/dp/B000065U30	Rating: 1.9047619047619047
http://www.amazon.com/dp/B000EQ5V86	Rating: 1.9689119170984455
http://www.amazon.com/dp/B000EQ5V8G	Rating: 1.9689119170984455
http://www.amazon.com/dp/B001ELC6FG	Rating: 2.1666666666666665
http://www.amazon.com/dp/B00005V9IO	Rating: 2.3333333333333335
http://www.amazon.com/dp/B000T12ILY	Rating: 2.64
http://www.amazon.com/dp/B00004RQBA	Rating: 2.6785714285714284
http://www.amazon.com/dp/B000VII4D6	Rating: 2.783505154639175
http://www.amazon.com/dp/B003JFEIJC	Rating: 2.84
http://www.amazon.com/dp/B002FLZKAM	Rating: 2.875

0.1 Spark and Pandas

```
[14]: from datetime import datetime
timeseries_rdd = reviews.map(lambda entry: {'score': entry['score'],
    ↪'time': datetime.fromtimestamp(entry['time'])})
```

```
[15]: import pandas as pd
import numpy as np
%matplotlib inline
import matplotlib.pyplot as plt
def parser(x):
```

```

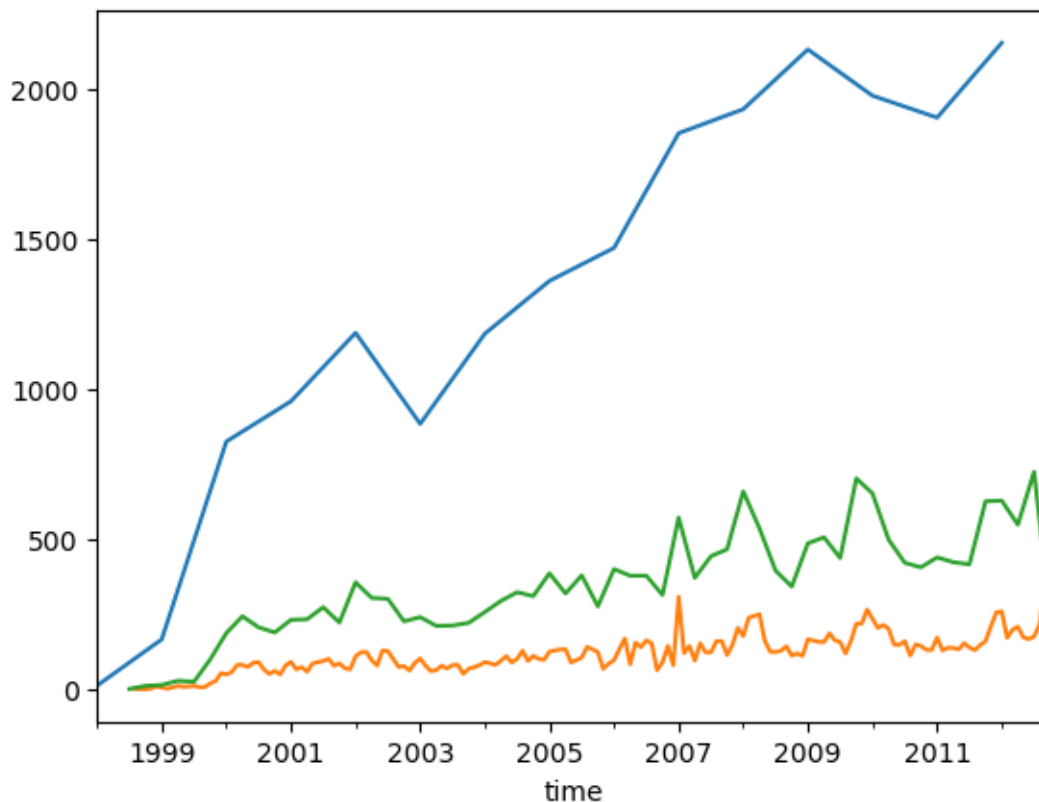
        return datetime.strptime('190'+x, '%Y-%m')
sample = timeseries_rdd.sample(withReplacement=False, fraction=20000.0/
    ↳ num_entries, seed=1134)
timeseries = pd.DataFrame(sample.collect(),
                           columns=['score', 'time'])
print(timeseries.head(3))
timeseries.score.astype('float64')
#timeseries.time.astype('datetime64')

timeseries.set_index('time', inplace=True)
Rsample = timeseries.score.resample('Y').count()
Rsample.plot()
Rsample2 = timeseries.score.resample('M').count()
Rsample2.plot()
Rsample3 = timeseries.score.resample('Q').count()
Rsample3.plot()

```

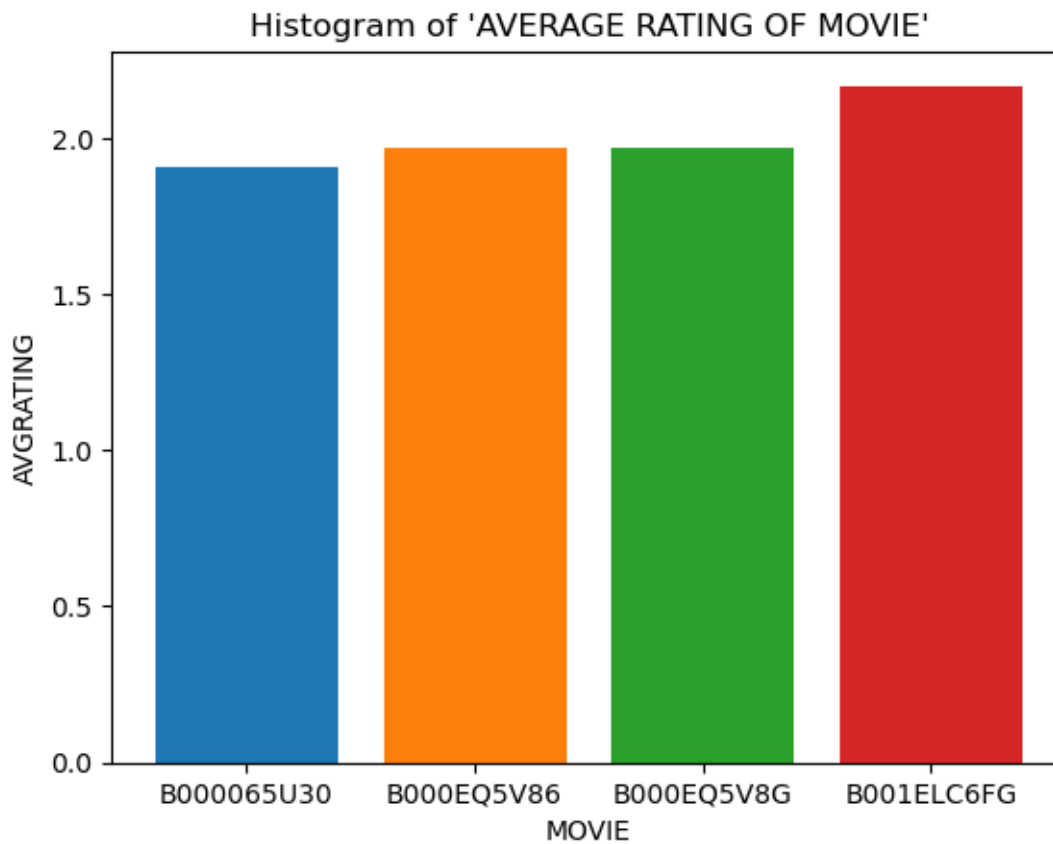
	score	time
0	5.0	2003-08-10 05:30:00
1	5.0	2003-08-18 05:30:00
2	5.0	2003-01-14 05:30:00

[15]: <Axes: xlabel='time'>

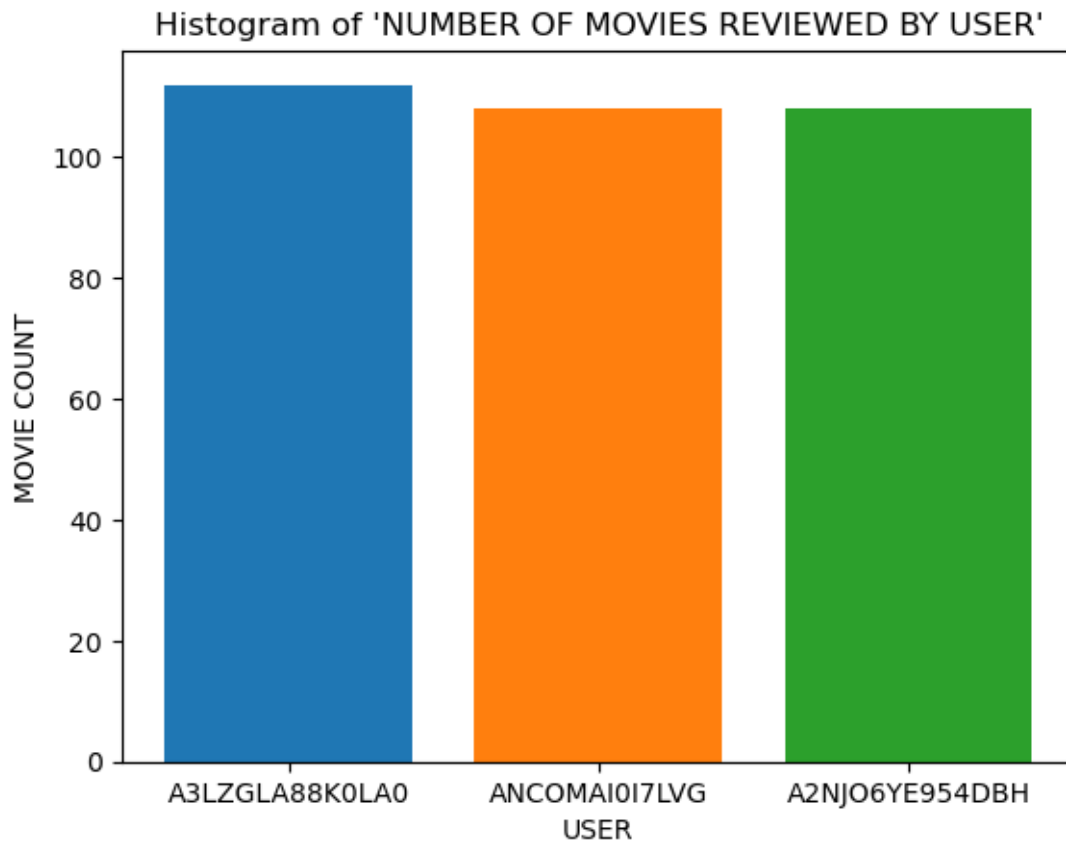


0.2 Matrix factorization

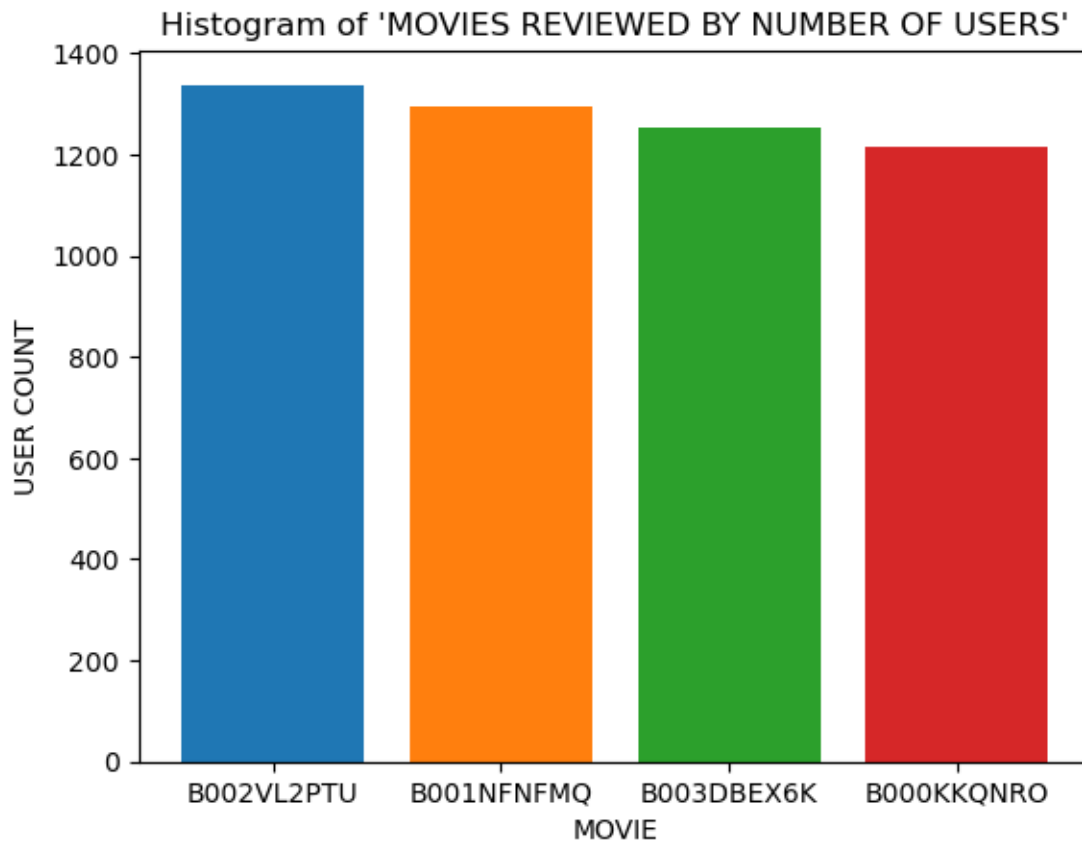
```
[16]: for movie in avg.take(4):  
    plt.bar(movie[1][0],movie[0][0])  
    plt.title('Histogram of \'AVERAGE RATING OF MOVIE\'')  
    plt.xlabel('MOVIE')  
    plt.ylabel('AVGRATING')
```



```
[17]: for movie in avg2.take(3):  
    plt.bar(movie[1][0],movie[0][0])  
    plt.title('Histogram of \'NUMBER OF MOVIES REVIEWED BY USER\'')  
    plt.xlabel('USER')  
    plt.ylabel('MOVIE COUNT')
```



```
[18]: for movie in avg3.take(4):  
    plt.bar(movie[1][0],movie[0][0])  
    plt.title('Histogram of \'MOVIES REVIEWED BY NUMBER OF USERS\'')  
    plt.xlabel('MOVIE')  
    plt.ylabel('USER COUNT')
```



```
[19]: from pyspark.mllib.recommendation import ALS
      from numpy import array
      import hashlib
      import math

      def get_hash(s):
          return int(hashlib.sha1(s).hexdigest(), 16) % (10 ** 8)

      #Input format: [user, product, rating]
      ratings = reviews.map(lambda entry: tuple([ get_hash(entry['user_id']).
          ↪encode('utf-8')),
                                                  get_hash(entry['product_id']).
          ↪encode('utf-8')),
                              int(entry['score']) ])

      train_data = ratings.filter(lambda entry: ((entry[0]+entry[1]) % 10) >= 2 )
      test_data  = ratings.filter(lambda entry: ((entry[0]+entry[1]) % 10) < 2 )
      train_data.cache()
```

```
#train_data.union(train_data)
print ("Number of train samples: " + str(train_data.count()))
print ("Number of test samples: " + str(test_data.count()))
```

Number of train samples: 39992

Number of test samples: 10008

```
[20]: # Build the recommendation model using Alternating Least Squares
from math import sqrt
rank = 20
numIterations = 20
model = ALS.train(train_data, rank, numIterations)
def convertToFloat(lines):
    returnedLine = []
    for x in lines:
        returnedLine.append(float(x))
    return returnedLine
# Evaluate the model on test data
unknown = test_data.map(lambda entry: (int(entry[0]),
    ↪int(entry[1])))
predictions = model.predictAll(unknown).map(lambda r: ((int(r[0]),
    ↪int(r[1])), r[2]))
true_and_predictions = test_data.map(lambda r: ((int(r[0]), int(r[1])), r[2])).
    ↪join(predictions)
MSE = true_and_predictions.map(lambda r: (int(r[1][0]) - int(r[1][1])**2).
    ↪reduce(lambda x, y: x + y)/true_and_predictions.count())
```

```
[21]: true_and_predictions.take(10)
```

```
[21]: [((77845900, 62577830), (5, 1.3422200314702162)),
      ((40572035, 58302865), (5, 2.8786042463025066)),
      ((40572035, 58302865), (5, 2.8786042463025066)),
      ((40572035, 58302865), (5, 2.8786042463025066)),
      ((40572035, 58302865), (5, 2.8786042463025066)),
      ((71882095, 58302865), (4, 0.46615785915578134)),
      ((8780925, 58302865), (5, 1.2592647672944959)),
      ((21981725, 65210615), (2, 1.4449015692952798)),
      ((35118635, 65210615), (3, 2.737150391175269)),
      ((61650965, 65210615), (3, 1.8188521500387471))]
```

0.3 No demo without a word count example!

```
[22]: min_occurrences = 10

good_reviews = reviews.filter(lambda line: line['score']==5.0)
bad_reviews = reviews.filter(lambda line: line['score']==1.0)
```

```

good_words      = good_reviews.flatMap(lambda line: line['review'].split(' '))
num_good_words  = good_words.count()

good_words = good_words.map(lambda word: (word.strip(), 1)) \
    .reduceByKey(lambda a, b: a+b) \
    .filter(lambda word_count: word_count[1] > min_occurrences)

bad_words = bad_reviews.flatMap(lambda line: line['review'].split(' '))
num_bad_words = bad_words.count()

bad_words = bad_words.map(lambda word: (word.strip(), 1)) \
    .reduceByKey(lambda a, b: a+b) \
    .filter(lambda word_count: word_count[1] > min_occurrences)

```

```

[23]: # Calculate the word frequencies
frequency_good = good_words.map(lambda word: ((word[0],), float(word[1])/
    ↪ num_good_words))
frequency_bad  = bad_words.map(lambda word: ((word[0],), float(word[1])/
    ↪ num_bad_words))

[24]: # Join the word frequencies of the good and bad reviews
joined_frequencies = frequency_good.join(frequency_bad)

[25]: # Calculate the relative difference of each word frequency in the good and bad
    ↪ reviews.
# Sort the dataset to get the most significant expressions for the
    ↪ characterization of either a positively
# or negatively rated movie.

import math

def relative_difference(a, b):
    return math.fabs(a-b)/a

result = joined_frequencies.map(lambda f: ((relative_difference(f[1][0],
    ↪ f[1][1]),), f[0][0]) ) \
    .sortByKey(ascending=False)

```

```

[26]: result.take(50)

```

```

[26]: [((31.56125203626583,), 'awful.'),
      ((30.02178160100973,), 'terrible.'),
      ((27.146167014399275,), 'Save'),
      ((26.694624228072016,), 'Kingsley'),
      ((26.052808695304964,), 'waste'),
      ((24.386738875732682,), 'redeeming'),

```



```

((22.179196364799406,), 'stupid.'),
((21.205280551152374,), 'crap.'),
((19.443763253425562,), 'worst'),
((18.26582554996314,), 'mess.'),
((15.73718594653048,), 'poorly'),
((14.728740390399597,), 'Bloodrayne'),
((14.452797576532936,), 'Kristanna'),
((13.569780572159626,), 'wasted'),
((13.449369162472356,), 'refund'),
((12.686763567786315,), 'wasting'),
((12.245255065599661,), 'Loken'),
((12.24525506559966,), 'ridiculous.'),
((12.038297955199667,), 'garbage'),
((11.803746563413005,), 'horrid'),
((11.669374410573587,), 'crappy'),
((11.548136377936519,), 'disappointment.'),
((11.381434083060553,), 'zero'),
((10.920729559039694,), 'bland'),
((10.82612059428541,), 'idiots'),
((10.686989763764407,), 'poor,'),
((10.528277557096,), 'predators'),
((10.479221056853039,), 'McG'),
((10.353075770513993,), 'pretentious'),
((10.037712554666383,), 'Poor'),
((10.037712554666383,), 'worthless'),
((10.037712554666383,), 'pyramid'),
((9.596204052479727,), 'bad.'),
((9.50485746582042,), 'insult'),
((9.301865051021958,), 'worse'),
((9.09162290712355,), 'trash'),
((8.933941299199745,), 'laughable'),
((8.933941299199745,), 'horrendous'),
((8.713187048106418,), 'worst.'),
((8.67922485563052,), 'poor.'),
((8.632912774981572,), 'travesty'),
((8.536583647231756,), 'sucks'),
((8.271678545919762,), 'Magdalene'),
((8.030855726545223,), '</>Unfortunately,'),
((7.987851651656912,), 'Stay'),
((7.960025485552711,), 'hack'),
((7.856766941455195,), 'awful'),
((7.787717303138237,), 'badly'),
((7.579312940217963,), 'boring,'),
((7.570459160093898,), 'prostitutes')]

```

```

[27]: for movie in result.take(7):
      plt.bar(movie[1],movie[0][0])

```

```
plt.title('Histogram of \'SENTIMENT ANALYSIS\'')
plt.xlabel('WORD')
plt.ylabel('NUMBER OF OCCURANCES')
```

